Teledyne Dalsa • 7075 Place Robert-Joncas, Suite 142 • St-Laurent, Quebec, H4M 2Z2 • Canada
www.teledynedalsa.com

X64-AN Quad™

User's Manual

Part number OC-64AM-USER0
Edition 2.01

NOTICE

Released on September 29, 2011

Document Number: OC-64AM-USER0

Printed in Canada

Contents

Introduction

Overview of the Manual

X64-AN Quad Board

- **The X64-AN Quad**

Description of the **X64-AN Quad** board, package contents, and a brief summary of its capabilities.

- **Block Diagrams**

Detailing **X64-AN Quad** functional blocks.

Installing the X64-AN Quad

- **The X64-AN Quad**

Description of the **X64-AN Quad** installation.

- **Configuring Sapera**

Describes Sapera servers and contiguous memory.

- **Troubleshooting**

Offers suggestions for resolving **X64-AN Quad** installation or usage problems.

Theory of Operation

- **Camera Control and Synchronization**

Overview of synchronization sources, trigger and strobe signals.

- **Acquisition Process**

Describes the process of video capture, from the analog input to the data in frame buffer, plus the events generated.

- **Trigger to Image Reliability**

Overview of the **X64-AN Quad** system designed for imaging confidence.

Technical Reference

- **X64-AN Quad** specifications including connector and pinout diagrams.

X-I/O Module Option

- Describes the X-I/O module, its configuration, cabling, and usage.

Sapera LT

- **Sapera Server and Parameters**

Lists the Sapera server available plus describes the Sapera parameters and values supported by **X64-AN Quad** board.

- **Using Sapera CamExpert with X64-AN Quad**

Describes the Sapera CamExpert tool and how to use it with **X64-AN Quad**.

- **Sapera Software Example**

Describes the Sapera Grab Demo example and how use it.

Support

- **Teledyne DALSA Contact Information**

Phone numbers, important web site links, and email addresses.

X64-AN Quad Board

Product Part Numbers

X64-AN Quad Board and Software	Product Number
X64-AN Quad with 128 MB of memory	OR-64A0-02040
Sapera LT version 6.30 or later (required but sold separately) 1. Sapera LT: Provides everything you will need to build your imaging application. Sapera 7.10 required for full feature support. 2. Current Sapera compliant board hardware drivers 3. Sapera documentation (compiled HTML help, and Adobe Acrobat® (PDF) formats)	OC-SL00-0000000
(optional) Sapera Processing Imaging Development Library (optional) includes: Over 600 optimized image processing routines	Contact Sales at Teledyne DALSA
(Optional) X-I/O Module (optional): provides 8 input & 8 output general I/Os (see "Appendix: X-I/O Module Option" on page 99 for information on the product and its interconnect cables)	OC-IO01-STD00
This manual, in printed form, is available on request	

X64-AN Quad Cables & Accessories	Product Number
Serial Ports / Strobe Outputs Connector Bracket Assembly supplied with each X64-AN Quad (connects to J19)	OR-64AC-0SER0
(optional) Trigger Input Cable – DB9 to four BNC	OR-VIPC-QDTRG
(optional) Power interface cable required when supplying power to cameras	OR-COMC-POW03
(optional) Hirose-12 Video Input Cable - 6 ft.	OC-COMC-HIR12

X64-AN Quad Overview

X64-AN Quad is a PCI-64 version 2.2 compatible plug-in board that provides image capture of up to 4 independent analog cameras. The acquisition circuitry interfaces with standard video (RS-170 and CCIR) and non-standard video (progressive scan cameras). X64-AN Quad interfaces with cameras easily with fully programmable standard Hirose-12 connectors.

X64-AN Quad provides an efficient 32/64-bit PCI interface, capable of bus mastering image data directly to memory within the system (i.e. system memory or another PCI target, such as VGA). Transfer rates up to 528MB/sec (64-bit PCI) or 100 MB/second (32-bit PCI) are sustained, depending upon host system capabilities. Consequently, images can be transferred to host memory in a fraction of the time acquired.

X64-AN Quad supports a number of event interrupt sources such as image acquisition and bus master transfer completion. X64-AN Quad contains 128MB of onboard memory for buffering image data between the camera and the host system. Onboard memory assures that image information is not lost during transfer to system memory due to PCI bus latency issues. Images are grabbed into local memory and then transferred at very high speeds to the host for processing or display.

X64-AN Quad is supported by Sapera LT. It is also fully supported by the Sapera Image Processing library.

X64-AN Quad Features

- Full size single slot PCI form factor
- 128MB onboard frame buffer memory
- Four (4) analog Hirose-12 video inputs; AC coupled and 75Ω terminated
- Acquires up to 4 monochrome cameras
- Supports standard RS-170 or CCIR and a variety of non-standard progressive scan cameras
- External Trigger input; synchronizes acquisition to external events
- Camera Control signals
- Strobe Control signals
- Resolution up to 4094 x 16,777,215 interlaced or non-interlaced
- Video controls allow brightness and contrast
- Driver supports 32-bit or 64-bit versions of Windows XP, Windows Vista and Windows 7
- 50MHz pixel clock

See "Technical Reference" on page 57 detailed information.

X64-AN Quad Functional Block Diagrams

Input Block Diagram (one shown)

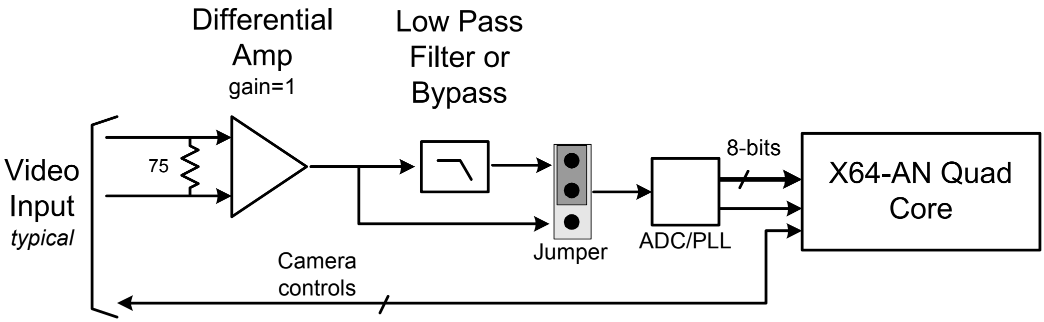


Figure 1: X64-AN Quad-Typical Input Block Diagram

- Four differential video inputs with high CMRR (typically 70dB).
- Input termination – 75 ohm.
- Wide band or low pass filtered signal path. Single corner low pass filter selected by shorting jumper. Filter specifications: analog 5th order Butterworth at 12.87 MHz.
- Independent programmable gain and offset references to adjust brightness and contrast of the analog image.
- One ADC (analog to digital converter) for each input.
- Each input has an independent clamper circuit and PLL (phase-locked loop).

Overall Block Diagram

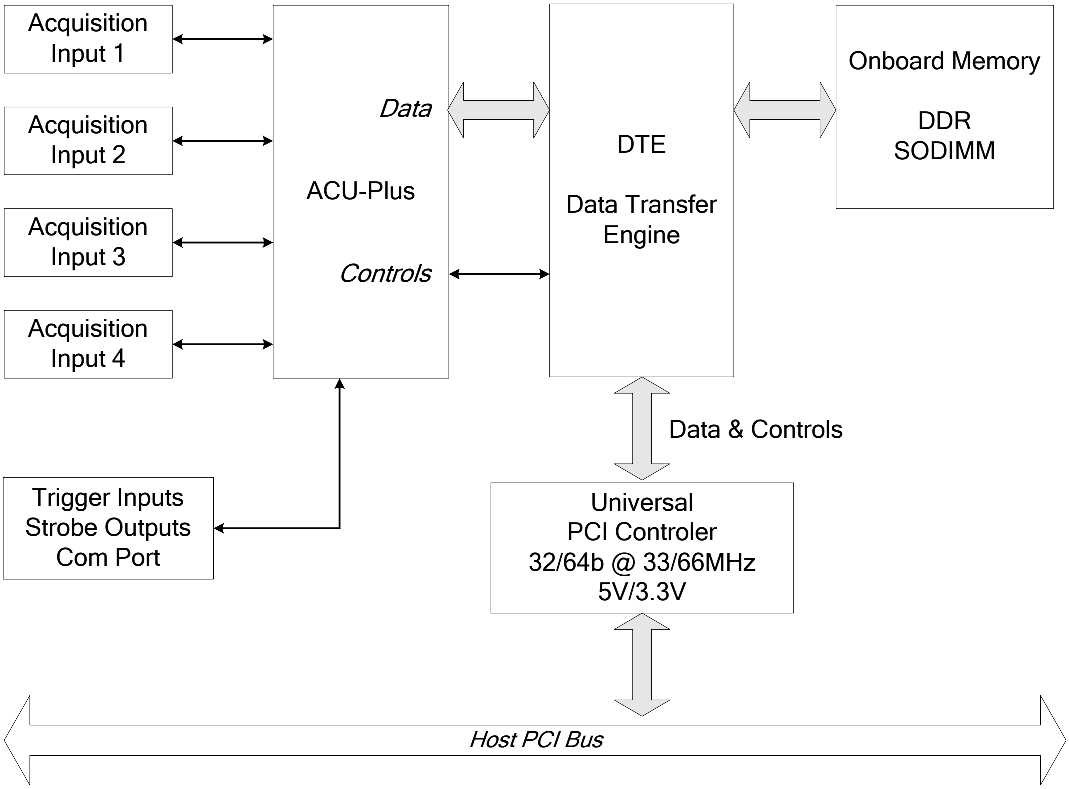


Figure 2: X64-AN Quad-General Block Diagram

- Acquisition inputs are described in "Input Block Diagram (one shown)" on page 5.

ACU-Plus: Acquisition Control Unit

The X64-AN Quad ACU consists of controller and functional logic as follows:

- Grab controller: includes sync control for slave cameras, trigger inputs/outputs, strobe outputs, etc.
- Serial port controller for camera setup.
- Memory manager for acquisition data to onboard memory and from memory to host system.
- Four independent 8-bit LUT for image processing.
- Cropper to manage region of interest operations.

DTE: Intelligent Data Transfer Engine

The X64-AN Quad intelligent Data Transfer Engine ensures fast image data transfers between the board and the host computer with zero CPU usage. The DTE provides a high degree of data integrity during continuous image acquisition in a non-real time operating system like Windows.

DTE consists of:

- Multiple independent DMA units
- Tap Descriptor Tables
- Auto-loading Scatter-Gather tables.

External Event Synchronization

To synchronize image captures with external events the X64-AN Quad features optically isolated trigger inputs and strobe control signals for each input channel. The signals can be programmed as active high or active low, edge or level based and can be controlled independently.

Development Software Overview

Sapera++ LT Library

Sapera++ LT is a powerful development library for image acquisition and control. Sapera++ LT provides a single API across all current and future Teledyne DALSA hardware. Sapera++ LT delivers a comprehensive feature set including program portability, versatile camera controls, flexible display functionality and management, plus easy to use application development wizards. Applications are developed using either C++ or .NET frameworks.

Sapera++ LT comes bundled with CamExpert, an easy to use camera configuration utility to create new, or modify existing camera configuration files.

Sapera Processing Library

Sapera Processing is a comprehensive set of C++ classes or .NET classes for image processing and analysis. Sapera Processing offers highly optimized tools for image processing, blob analysis, search (pattern recognition), OCR and barcode decoding.

About the X-I/O Module

The optional X-I/O module adds general purpose software controllable I/O signals to the X64-AN Quad. The X-I/O module provides 2 opto-coupled inputs, 6 logic signal inputs (5V or 24V), and 8 TTL outputs (NPN or PNP type selectable). The module also makes available 5V or 12V dc power from the host system.

The X-I/O module can be either purchased with the X64-AN Quad board or installed into the computer system at a later time. The module occupies one adjacent PCI slot and connects to the X64-AN Quad via a ribbon cable. X-I/O Module external connections are made via the DB37 connector on the module bracket.

X-I/O requires X64-AN Quad board driver version 1.10 or later and Sapera LT version 5.30 or later.

See "Appendix: X-I/O Module Option" on page 99 for details and specifications.

Installing the X64-AN Quad

Warning! (Grounding Instructions)

Static electricity can damage electronic components. Please discharge any static electrical charge by touching a grounded surface, such as the metal computer chassis, before performing any hardware installation.

If you do not feel comfortable performing the installation, please consult a qualified computer technician.

Important: Never remove or install any hardware component with the computer power on. Disconnect the power cord from the computer to disable the power standby mode. This prevents the case where some computers unexpectedly power up on installation of a board.

Installation

Note: to install Sapera LT and the X64-AN Quad device driver, logon to the workstation as administrator or with an account that has administrator privileges.

The Sapera LT Development Library (or 'runtime library' if application execution without development is preferred) must be installed before the board device driver.

- Turn the computer off, disconnect the power cord (disables power standby mode), and open the computer chassis to allow access to the expansion slot area.
- Install the X64-AN Quad into a free 64-bit PCI expansion slot. If no 64-bit PCI slot is available, use a common 32-bit PCI slot. X64-AN Quad supports the plug and play automatic configuration of the PCI specification.
- Connect the J17 12V power connector to a floppy power cable using the optional floppy power connector (OC-COMC-PCPWR). See "X64-AN Quad Connector and Jumper Locations" on page 61 for detailed descriptions.
- Close the computer chassis and turn the computer on.
- Windows will find the X64-AN Quad and start its **Found New Hardware Wizard**. Click on the **Cancel** button to close the Wizard Application.

- If using **Windows Vista or Windows 7**, Windows will display its **Found New Hardware dialog**. Click on the default "*Ask me again later*" and continue with the installation. Note that if you select the third option "*Don't show this message again for this device*", there will be no prompt if the Teledyne DALSA board is installed in the same computer.

Sapera LT Library Installation

- Insert the Teledyne DALSA Sapera Essential CD-ROM. With **AUTORUN** enabled, the installation menu automatically displays.
- With **AUTORUN** not enabled, use Windows Explorer and browse to the root directory of the CD-ROM. Execute **launch.exe** to start the installation menu and install the required Sapera components.
- Continue with the installation of the board driver as described in the next section.
- The installation program will prompt you to reboot the computer.

Refer to *Sapera LT User's Manual* for additional details about Sapera LT.

X64-AN Quad Driver Installation

The X64-AN Quad board driver supports installation in a Windows XP, Windows Vista, or Windows 7 system.

- After installing Sapera, continue by selecting the X64-AN Quad driver installation.
- If Sapera was previously installed, insert the Teledyne DALSA Sapera Essential CD-ROM to install the board driver. With **AUTORUN** enabled, the installation menu automatically displays. Install the X64-AN Quad driver.
- With **AUTORUN** not enabled, use Windows Explorer and browse to the root directory of the CD-ROM. Execute **launch.exe** to start the installation menu and install the X64-AN Quad driver. During the late stages of the installation, the X64-AN Quad firmware loader application starts. See the description in the following section.
- If Windows displays any unexpected message concerning the installed board, power off the system and verify that the X64-AN Quad is installed properly in the computer slot.

X64-AN Quad Firmware Loader

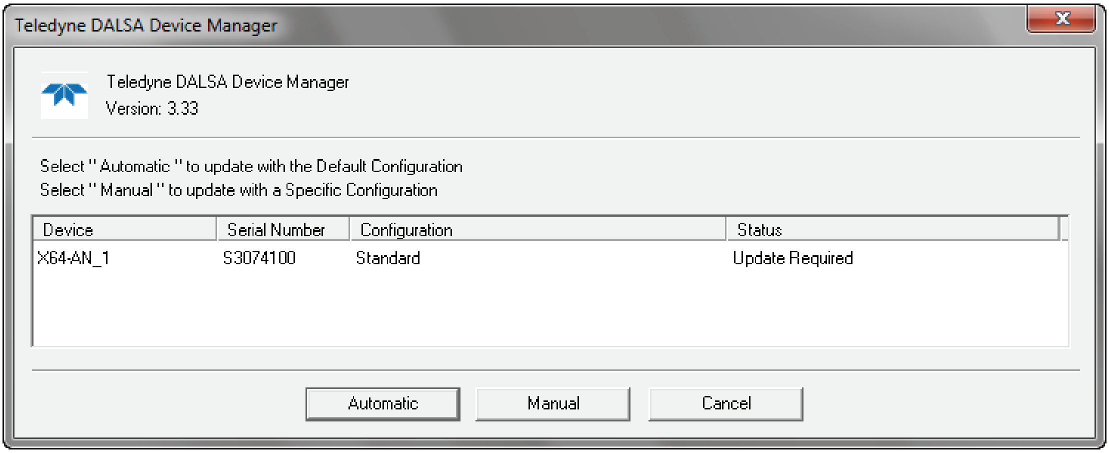
The Device Manager-Firmware Loader program automatically executes at the end of the driver installation and on every subsequent reboot of the computer. It will determine if the X64-AN Quad requires a firmware update. If firmware is required, a dialog displays. This dialog also allows the user to load firmware for alternate operational modes of the board.

Important: In the vary rare case of firmware loader errors please see "Recovering from a Firmware Update Error" on page 28.

Firmware Update: Automatic Mode

Click **Automatic** to update the X64-AN Quad firmware with the default functionality as supported by the X64-AN Quad hardware. If there are multiple X64-AN Quad boards in the system, all boards will be updated with new firmware.

With multiple X64-AN Quad boards in the system, all are updated with new firmware. If any installed board in a system already has the correct firmware version, an update is not required. In the following screen shot, a single board is installed and ready for a firmware upgrade.



Teledyne DALSA Device Manager

Teledyne DALSA Device Manager
Version: 3.33

Select " Automatic " to update with the Default Configuration
Select " Manual " to update with a Specific Configuration

Device	Serial Number	Configuration	Status
X64-AN_1	S3074100	Standard	Update Required

Figure 3: X64-AN Quad Firmware Loader

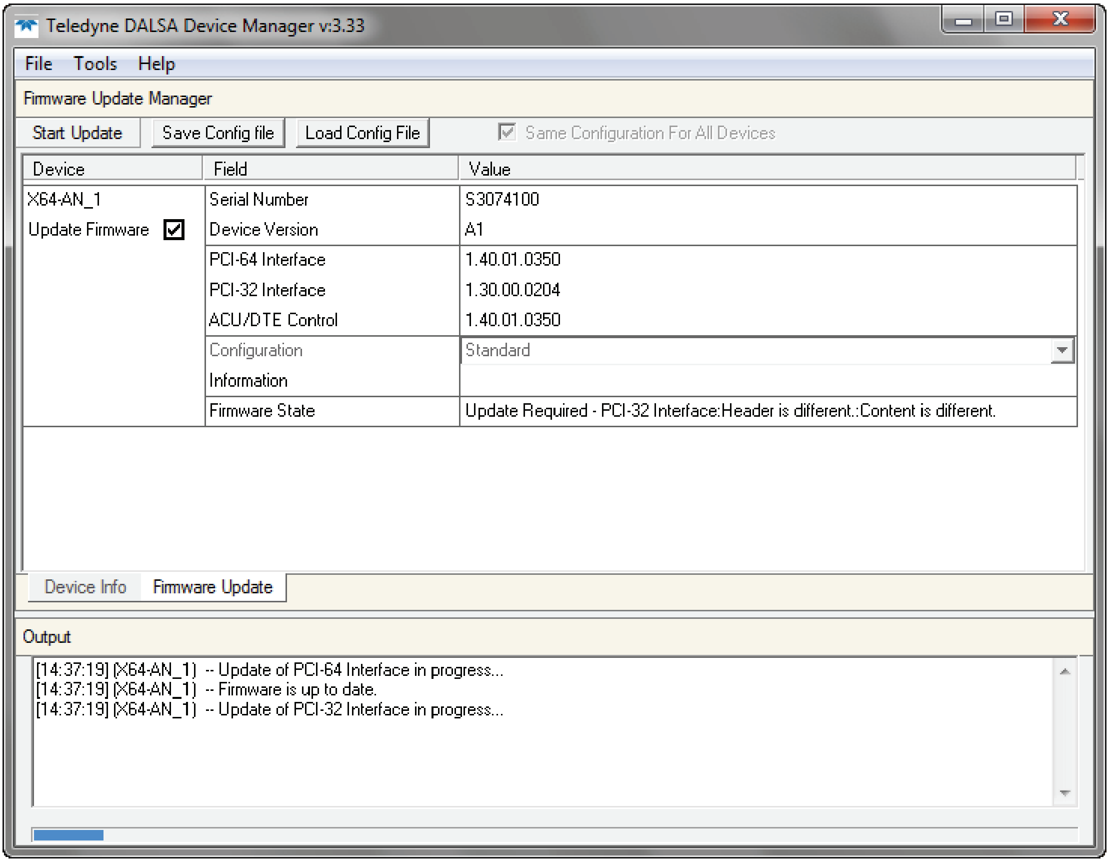
Note: if you are certain that the X64-AN Quad firmware is of the same version as the driver being installed, you can click on **Cancel** to bypass the update procedure.

Firmware Update: Manual Mode

Select **Manual** mode to load firmware other then the default version. The figure below shows the Device Manager manual firmware screen (with one X64-AN Quad installed in the system). Information on all installed X64-AN Quad boards, their serial numbers, and their firmware components are shown.

Do a manual firmware update is as follows:

- Select the X64-AN Quad via the selection box (if there are multiple boards in the system).
- From the Configuration field drop menu select the firmware version required.
- Click on the Start Update button.
- Observe the firmware update progress in the message output window.
- Close the Device manager program when the X64-AN Quad board reset complete message is shown.



Device	Field	Value
X64-AN_1	Serial Number	S3074100
Update Firmware ☑	Device Version	A1
	PCI-64 Interface	1.40.01.0350
	PCI-32 Interface	1.30.00.0204
	ACU/DTE Control	1.40.01.0350
	Configuration	Standard
	Information	
	Firmware State	Update Required - PCI-32 Interface:Header is different.:Content is different.

```
[14:37:19] [X64-AN_1]  -- Update of PCI-64 Interface in progress...
[14:37:19] [X64-AN_1]  -- Firmware is up to date.
[14:37:19] [X64-AN_1]  -- Update of PCI-32 Interface in progress...
```

Figure 4: X64-AN Quad Firmware Loader while programming

Firmware versions currently available are:

- **Standard:**
  Support for 4 independent monochrome cameras, with a pixel clock range of 8 MHz to 50 MHz.

Note: Also choose the manual firmware upgrade mode when multiple X64-AN Quad boards are installed in the system and if each requires different firmware versions.

Executing the Firmware Loader from the Start Menu

If required, the X64-AN Quad Firmware Loader program is executed via the Windows Start Menu shortcut **Start • All Programs • Teledyne DALSA • X64-AN Quad Device Driver • Firmware Update**.

Requirements for a Silent Install

Both Sapera LT and the X64-AN Quad driver installations share the same installer technology. When the installations of Teledyne DALSA products are embedded within a third party's product installation, the mode can either have user interaction or be completely silent. The following installation mode descriptions apply to both Sapera and the hardware driver.

Note: You must reboot after the installation of Sapera LT. However, to streamline the installation process, Sapera LT can be installed without rebooting before installing the board hardware device drivers. The installations then complete with a single final system reboot.

Perform Teledyne DALSA embedded installations in either of these two ways:

- **Normal Mode**

The default mode is interactive. This is identical to running the setup.exe program manually from Windows (either run from Windows Explorer or the Windows command line).

- **Silent Mode**

This mode requires no user interaction. A preconfigured "response" file provides the user input. The installer displays nothing.

Silent Mode Installation

A Silent Mode installation is recommended when integrating Teledyne DALSA products into your software installation. The silent installation mode allows the device driver installation to proceed without the need for mouse clicks or other input from a user.

Preparing a Silent Mode Installation requires two steps:

- Prepare the response file, which emulates a user.
- Invoke the device driver installer with command options to use the prepared response file.

Creating a Response File

Create the installer response file by performing a device driver installation with a command line switch "-r". The response file is automatically named **`setup.iss`** and is saved in the \windows folder. If a specific directory is desired, the switch –f1 is used.

As an example, to save a response file in the same directory as the installation executable of the X64-AN Quad, the command line would be:

```
X64-AN_Quad_1.40.00.0000 -r -f1".\setup.iss"
```

Running a Silent Mode Installation

A device driver silent installation, whether done alone or within a larger software installation requires the device driver executable and the generated response file **`setup.iss`**.

Execute the device driver installer with the following command line:

```
X64-AN_Quad_1.40.00.0000 -s -f1".\setup.iss"
```

Where the **–s** switch specifies the silent mode and the **–f1** switch specifies the location of the response file. In this example, the switch –f1".\setup.iss" specifies that the **`setup.iss`** file be in the same folder as the device driver installer.

ⓘ	**Note:** On Windows Vista and 7, the Windows Security dialog box will appear unless one has already notified Windows to ‘Always trust software from “DALSA Corp.” during a previous installation of a driver.

Silent Mode Uninstall

Similar to a silent installation, a response file must be prepared first as follows.

Creating a Response File

The installer response file is created by performing a device driver un-installation with a command line switch "-r". The response file is automatically named **`setup_uninstall.iss`** which is saved in the \windows folder. If a specific directory is desired, the switch “–f1” is used.

As an example, to save a response file in the same directory as the installation executable of the X64-AN Quad, the command line would be:

```
X64-AN_Quad_1.40.00.0000 -r –f1”.\setup_uninstall.iss”
```

Running a Silent Mode Uninstall

Similar to the device driver silent mode installation, the un-installation requires the device driver executable and the generated response file **`setup.iss`**.

Execute the device driver installer with the following command line:

```
X64-AN_Quad_1.40.00.0000 -s -f1".\setup_uninstall.iss"
```

Where the **–s** switch specifies the silent mode and the **–f1** switch specifies the location of the response file. In this example, the switch –f1".\setup_uninstall.iss" specifies that the **`setup_uninstall.iss`** file be in the same folder as the device driver installer.

Silent Mode Installation Return Code

A silent mode installation creates a file "corinstall.ini" in the Windows directory. A section called [SetupResult] contains the 'status' of the installation. A value of 1 indicates that the installation has started and a value of 2 indicates that the installation has terminated.

A silent mode installation also creates a log file "setup.log" which by default is created in the same directory and with the same name (except for the extension) as the response file. The /f2 option enables you to specify an alternative log file location and file name, as in Setup.exe /s /f2"C:\Setup.log".

The "setup.log" file contains three sections. The first section, [InstallShield Silent], identifies the version of InstallShield used in the silent installation. It also identifies the file as a log file. The second section, [Application], identifies the installed application name, version, and the company name. The third section, [ResponseResult], contains the 'ResultCode' indicating whether the silent installation succeeded. A value of 0 means the installation was successful.

Installation Setup with CorAppLauncher.exe

The installation setup can be run with the CorAppLauncher.exe tool provided with the driver.

- Install the board driver and get CorAppLauncher.exe from the \bin directory of the installation.
- When running the installation, CorAppLauncher.exe will return only when the installation is finished.
- When run from within a batch file, obtain the installation exit code from the ERRORLEVEL value.
- The arguments to CorAppLauncher.exe are
  -l: Launch application
  -f: Application to launch. Specify a fully qualified path.

As an example:

- CorAppLauncher –l –f'c:\driver_install\x64-AN_Quad_1.40.00.0000.exe"
- IF %ERRORLEVEL% NEQ 0 goto launch error

Note: There is a 32-bit and 64-bit version of CorAppLauncher.exe. When installing the driver, only the version related to the OS is installed. However, the 32-bit version is usable on either 32-bit or 64-bit Windows.

Custom Driver Installation using install.ini

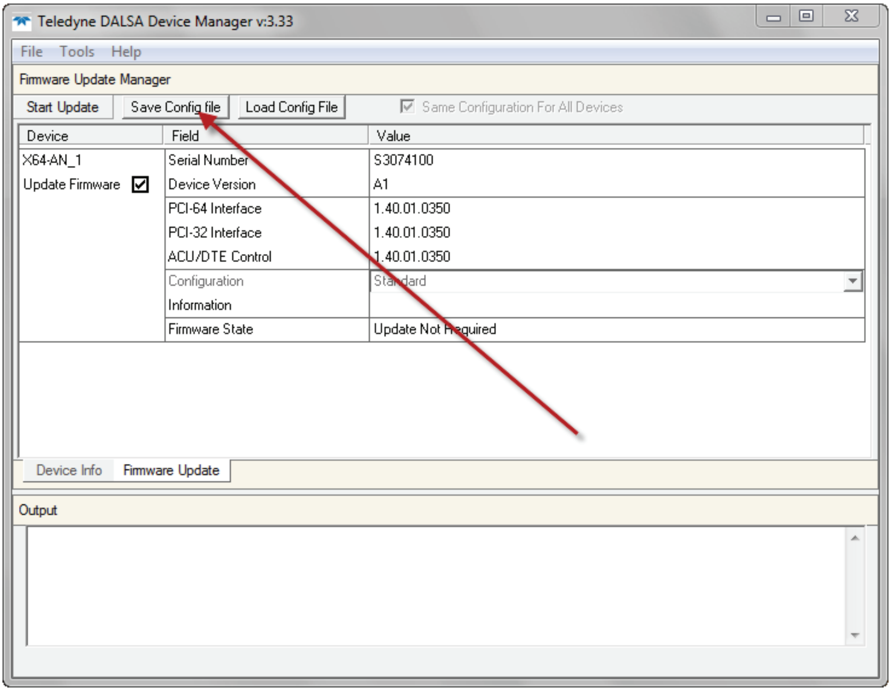
Customize the driver installation by parameters defined in the file "install.ini". By using this file, the user can:

- Select the user default configuration.
- Select different configurations for systems with multiple boards.

- Assign a standard Serial COM port to board.

Creating the install.ini File

- Install the driver in the target computer. All X64-AN Quad boards required in the system must be installed.
- Configure each board's acquisition firmware using the Teledyne DALSA Device Manager tool (see Device Manager – Board Viewer).
- If a standard Serial COM port is required for any board, use the Sapera Configuration tool (see COM Port Assignment).
- When each board setup is complete, using the Teledyne DALSA Device Manager tool, click on the Save Config File button. This will create the "install.ini" file.



Device	Field	Value
X64-AN_1	Serial Number	S3074100
Update Firmware ☑	Device Version	A1
	PCI-64 Interface	1.40.01.0350
	PCI-32 Interface	1.40.01.0350
	ACU/DTE Control	1.40.01.0350
	Configuration	Standard
	Information	
	Firmware State	Update Not Required

Figure 5: Create an install.ini File

Run the Installation using install.ini

Copy the install.ini file into the same directory as the setup installation file. Run the setup installation as normal. The installation will automatically check for an install.ini file and if found, use the configuration defined in it.

Upgrading Sapera or any Teledyne DALSA Board Driver

When installing a new version of Sapera or a Teledyne DALSA acquisition board driver in a computer with a previous installation, the **current version must be un-installed first**. Upgrade scenarios are described below. Note that if the board is installed in a different slot, the new hardware wizard opens. Answer as instructed in section "X64-AN Quad Driver Installation" on page 10.

Board Driver Upgrade Only

Minor upgrades to acquisition board drivers are typically distributed as ZIP files available in the Teledyne DALSA web site www.teledynedalsa.com/mv/support. Board driver revisions are also available on the next release of the Sapera Essential CD-ROM.

Often minor board driver upgrades do not require a new revision of Sapera. To confirm that the current Sapera version will work with the new board driver:

- Check the new board driver ReadMe file before installing, for information on the minimum Sapera version required.
- If the ReadMe file does not specify the Sapera version required, you should contact Teledyne DALSA Technical Support (see "Technical Support" on page 114).

To upgrade the board driver only:

- Logon the computer as an administrator or with an account that has administrator privileges.
- In **Windows XP**, from the start menu select **Start • Settings • Control Panel • Add or Remove Programs**. Select the Teledyne DALSA X64-AN Quad board driver and click **Remove**.

Windows XP only:

- When the driver un-install is complete, reboot the computer.
- Logon the computer as an administrator again.

- In **Windows Vista and Windows 7**, from the start menu select **Start • Settings • Control Panel • Programs and Features**. Double-click the Teledyne DALSA X64-AN Quad board driver and click **Remove**.
- Install the new board driver. Run **Setup.exe** if installing manually from a downloaded driver file.
- If the new driver is on a Sapera Essential CD-ROM follow the installation procedure described in "Installation" on page 9.
- **Important:** you can not install a Teledyne DALSA board driver without Sapera LT installed on the computer.

Upgrading both Sapera and Board Driver

When both Sapera LT and the Teledyne DALSA acquisition board driver are upgraded, follow the procedure described below.

- Logon the computer as an administrator or with an account that has administrator privileges.
- In **Windows XP**, from the start menu select **Start • Settings • Control Panel • Add or Remove Programs**. Select the Teledyne DALSA X64-AN Quad board driver and click **Remove**. Follow by also removing the older version of Sapera LT.
- In **Windows Vista and Windows 7**, from the start menu select **Start • Settings • Control Panel • Programs and Features**. Double-click the Teledyne DALSA X64-AN Quad board driver and click **Remove**. Follow by also removing the older version of Sapera LT.
- Reboot the computer and logon the computer as an administrator again.
- Install the new versions of Sapera and the board driver as if this was a first time installation. See "Sapera LT Library Installation" on page 10 and "Installation" on page 9 for installation procedures.

Connecting Camera and Devices

Connector Bracket End View

The following figure identifies the four X64-AN Quad input connectors. The Hirose-12 connectors have programmable signal pins, defined by camera file parameters. See "Hirose Input Connectors" on page 63 for pinout descriptions. The DB9 connector J5 provides for easy trigger signal connections. See "J5 – Trigger Signals Connector" on page 65 for pinout descriptions.

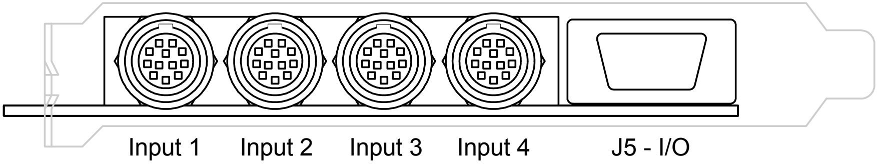


Figure 6: X64-AN Quad Connector Bracket

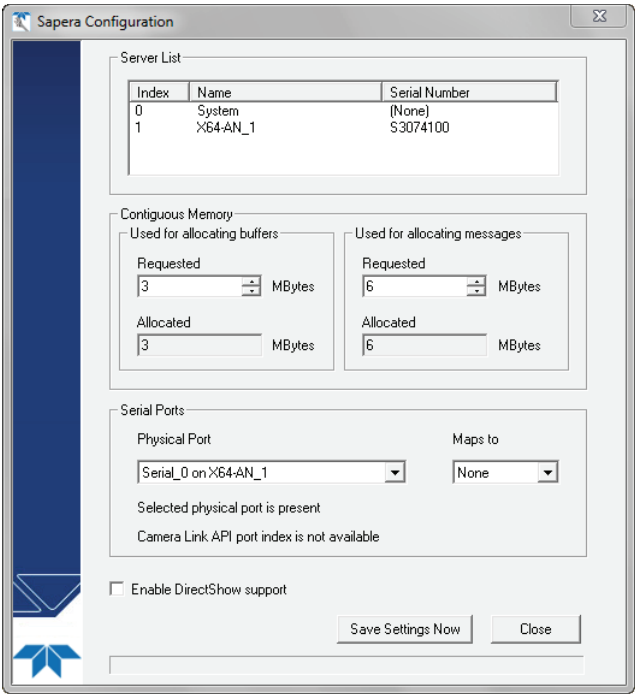
Note: An X64-AN Quad board is able to provide up to 500mA of power to connected cameras from the PCI bus (fused protected). Nonetheless, Teledyne DALSA strongly recommends connecting the floppy power connector (see "J17: Power Connector" on page 69) to ensure sufficient current is available from the PC power supply.
Caution: Sinking more than 500mA from the X64-AN Quad PCI connector may result in the auto-reset fuse blowing. Check your camera datasheet for the required camera current.

Configuring Sapera

The Sapera Configuration program (**Start • All Programs • Teledyne DALSA • Sapera LT • Sapera Configuration**) allows the user to see all available Sapera servers for the installed Sapera-compatible boards.

Viewing Installed Sapera Servers

The **System** entry represents the system server. It corresponds to the host machine (your computer) and is the only server that should be present at all times. As shown in the following screen image, server index 1 is the X64-AN Quad board installed.



Sapera Configuration

Server List

Index	Name	Serial Number
0	System	(None)
1	X64-AN_1	S3074100

Contiguous Memory

Used for allocating buffers
Requested: 3 MBytes
Allocated: 3 MBytes

Used for allocating messages
Requested: 6 MBytes
Allocated: 6 MBytes

Serial Ports

Physical Port: Serial_0 on X64-AN_1
Maps to: None

Selected physical port is present

Camera Link API port index is not available

□ Enable DirectShow support

Figure 7: Sapera configuration utility

Increasing Contiguous Memory for Sapera Resources

The **Contiguous Memory** section lets the user specify the total amount of contiguous memory (a block of physical memory occupying consecutive addresses) reserved for the resources needed for **Sapera buffers** allocation and **Sapera messaging**. For both items, the **Requested** value dialog box shows the driver default memory setting while the **Allocated** value displays the amount of contiguous memory that has been allocated successfully. The default values will generally satisfy the needs of most applications.

The **Sapera buffers** value determines the total amount of contiguous memory reserved at boot time for the allocation of dynamic resources used for frame buffer management, such as the scatter-gather list, DMA descriptor tables, plus other kernel needs. Adjust this value higher if your application generates any out-of-memory error while allocating host frame buffers or when connecting the buffers via a transfer object. You can approximate the amount of contiguous memory required as follows:

- Calculate the total amount of host memory used for frame buffers
[number of frame buffers • number of pixels per line • number of lines • (2 - if buffer is 10 or 12 bits)].
- Provide 1MB for every 256 MB of frame buffer memory required.
- Add an additional 1 MB if the frame buffers have a short line length, say 1k or less
(the increased number of individual frame buffers requires more resources).
- Add an additional 2 MB for various static and dynamic Sapera resources.
- Add the amount of memory needed for DMA tables using the formula (Sapera 7.10 and up):
[number of frame buffers • number of lines • 16 • (line length in bytes / 4kB)].
- Test for any memory error when allocating host buffers. Simply use the Buffer menu of the Sapera Grab demo program (see "Using the Grab Demo" on page 96) to allocate the number of host buffers required for your acquisition source. Feel free to test the maximum limit of host buffers possible on your host system – the Sapera Grab demo will not crash when the requested number of host frame buffers is not allocated.

Host Computer Frame Buffer Memory Limitations

When planning a Sapera application and its host frame buffers used, as well as other Sapera memory resources, do not forget the Windows operating system memory needs.

A Sapera application using the preferred *scatter gather buffers* could consume most of the remaining system memory, with a large allocation of frame buffers. If using frame buffers allocated as a *single contiguous memory block*, Windows will limit the allocation dependent on the installed system memory. Use the Buffer menu of the Sapera Grab demo program to allocate host buffer memory until an error message signals the limit allowed by the operating system used.

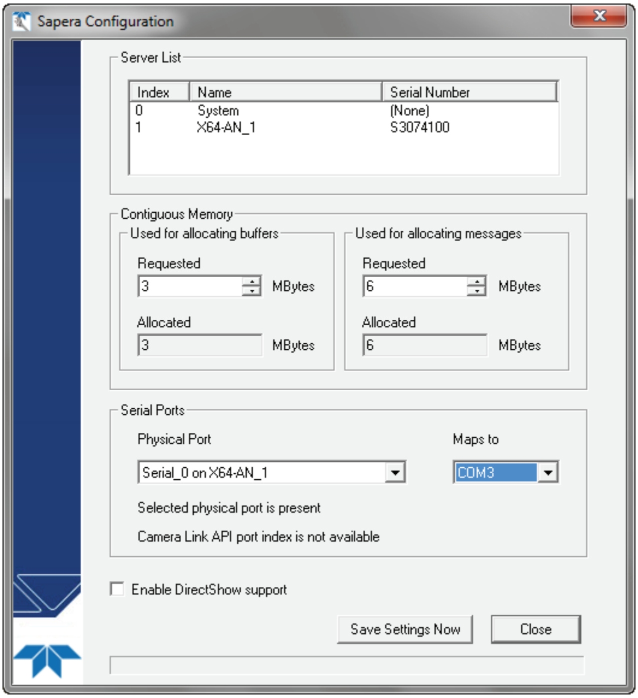
Contiguous Memory for Sapera Messaging

The current value for **Sapera messaging** determines the total amount of contiguous memory reserved at boot time for message allocation. This memory space is used to store arguments when a Sapera function is called. Increase this value if you are using functions with large arguments, such as arrays and when experiencing any memory errors.

COM Port Assignment

The lower section of the Sapera Configuration program screen contains the serial port configuration menu. Configure as follows:

- Use the **Physical Port** drop menu to select the Sapera board device from all available Sapera boards with serial ports (when more then one board is in the system). Note that the X64-AN Quad has 4 physical ports.
- Use the **Maps to** drop menu to assign an available COM number to the Sapera board serial port.
- Click on the **Save Settings Now** button and then the **Close** button. You are prompted to reboot your computer to enable the serial port mapping.
- The X64-AN Quad serial ports (one of them mapped to COM3 in this example) is available to any serial port application for camera control. Note that this serial port is not listed in the **Windows • Control Panel • System Properties • Device Manager** because it is a logical serial port mapping.



Sapera Configuration

Server List

Index	Name	Serial Number
0	System	(None)
1	X64-AN_1	S3074100

Contiguous Memory

Used for allocating buffers — Requested: 3 MBytes; Allocated: 3 MBytes

Used for allocating messages — Requested: 6 MBytes; Allocated: 6 MBytes

Serial Ports

Physical Port: Serial_0 on X64-AN_1

Maps to: COM3

Selected physical port is present

Camera Link API port index is not available

☐ Enable DirectShow support

Figure 8: Serial port mapping

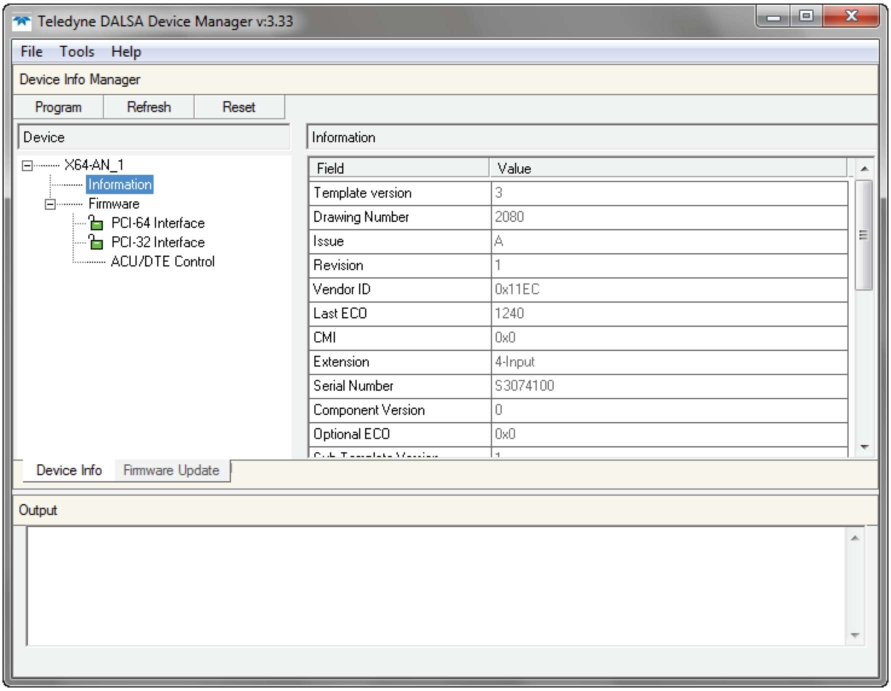
Displaying X64-AN Quad Information

The Device Manager program also displays information about the X64-AN Quad boards installed in the system. To view board information run the program via the Windows Start Menu shortcut **Start • All Programs • Teledyne DALSA • X64-AN Quad Device Driver • Device Manager**.

Device Manager – Board Viewer

The following screen image shows the Device Manager program with the Information/Firmware tab active. The left window displays all Teledyne DALSA boards in the system and their individual device components. The right window displays the information stored in the selected board device. This example screen shows the X64-AN Quad information contained in the EEProm component.

Generate the device manager report file (BoardInfo.txt) by clicking **File • Save Device Info**. Teledyne DALSA Technical Support may request this report to aid in troubleshooting installation or operational problems.



Field	Value
Template version	3
Drawing Number	2080
Issue	A
Revision	1
Vendor ID	0x11EC
Last ECO	1240
CMI	0x0
Extension	4-Input
Serial Number	S3074100
Component Version	0
Optional ECO	0x0
Sub Template Version	1

Figure 9: Board Information via Device Manager

Troubleshooting Problems

Overview

The X64-AN Quad has been tested by Teledyne DALSA in a wide variety of 64-bit and 32-bit PCI computers. Although unlikely, installation problems may occur due to the constant changing nature of computer equipment and operating systems. This section describes what the user can verify to determine the problem or the checks to make before contacting Teledyne DALSA Technical Support.

If you require help and need to contact Teledyne DALSA Technical Support, make detailed notes on your installation and/or test results for our technical support to review. See "Technical Support" on page 114 for contact information.

Problem Type Summary

X64-AN Quad problems are either installation types where the board hardware is not recognized on the PCI bus, or function errors due to camera connections or bandwidth issues. The following links jump to various topics in this troubleshooting section.

First Step: Check the Status LEDs

The four input status LEDs located on the top edge of the board, should be RED when the board is initialized properly but with no video signals connected. Other LED indicators relate to various input signal states as described in section "Acquisition Status LED" on page **67**.

Possible Installation Problems

- **Hardware PCI bus conflict:** When a new installation produces PCI bus error messages or the board driver does not install, it is important to verify that there are no conflicts with other PCI or system devices already installed. Use the Teledyne DALSA PCI Diagnostic tool as described in "Checking for PCI Bus Conflicts" on page 25. Also verify the installation via the "Windows Device Manager" on page 26.
- **Verify Sapera and Board drivers:** If there are errors when running applications, confirm that all Sapera and board drivers are running. See "Sapera and Hardware Windows Drivers" on page 27 for details. In addition, Teledyne DALSA technical support will ask for the log file of messages by Teledyne DALSA drivers. Follow the instructions describe in "Teledyne DALSA Log Viewer" on page 30.
- **Firmware update error:** There was an error during the X64-AN Quad firmware update procedure. The user usually easily corrects this. Follow the instructions "Recovering from a Firmware Update Error" on page 28.
- Installation went well but the board doesn't work or stopped working. Review theses steps described in "Symptoms: CamExpert Detects no Boards" on page 31.

Possible Functional Problems

- **Driver Information:** Use the Teledyne DALSA device manager program to view information about the installed X64-AN Quad board and driver. See "Driver Information via the Device Manager Program" on page 29.
- **Area Scan Memory Requirements:** The X64-AN Quad on board memory provides two frame buffers large enough for most imaging situations. See "Memory Requirements with Area Scan Acquisitions" on page 30 for details on the on board memory and possible limitations.

Sometimes the problem symptoms are not the result of an installation issue but due to other system issues. Review the sections described below for solutions to various functional problems.

- "Symptoms: X64-AN Quad Does Not Grab" on page 31
- "Symptoms: Card grabs black" on page 32
- "Symptoms: Card acquisition bandwidth is less than expected" on page 32

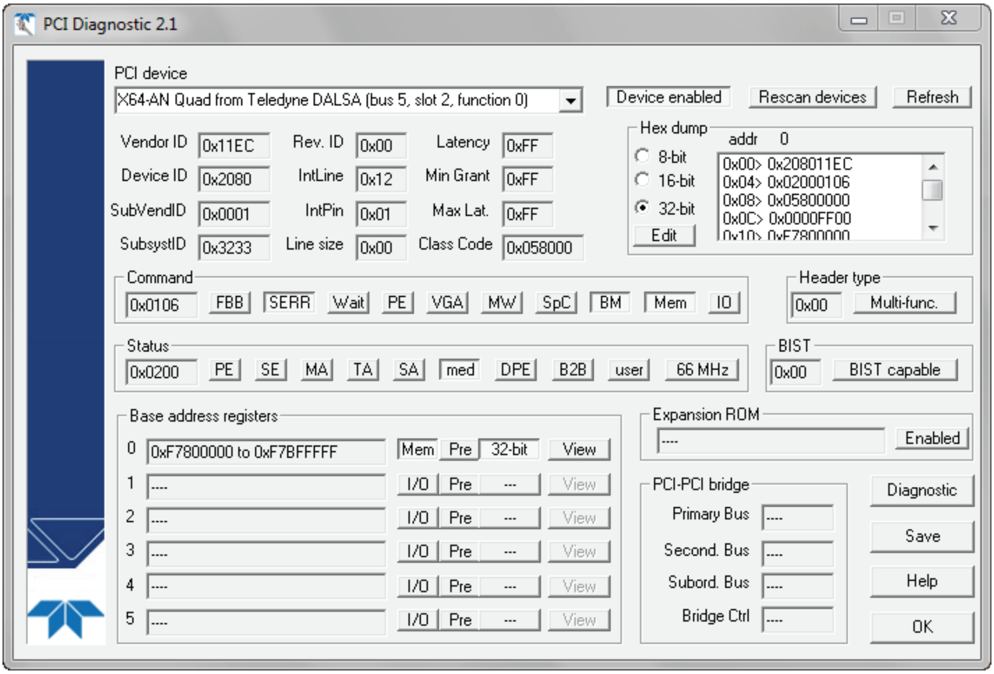
Troubleshooting Procedures

The following sections provide information and solutions to possible X64-AN Quad installation and functional problems. The previous section of this manual summarizes these topics.

Checking for PCI Bus Conflicts

One of the first items to check when there is a problem with any PCI board is to examine the system PCI configuration and ensure that there are no conflicts with other PCI or system devices. The Teledyne DALSA *PCI Diagnostic* program (**cpcidiag.exe**) allows examination of the PCI configuration registers and can save this information to a text file. Run the program via the Windows Start Menu shortcut **Start • All Programs • Teledyne DALSA • Sapera LT • Tools • PCI Diagnostics**.

As shown in the following screen image, use the first drop menu to select the PCI device to examine. Select the device "X64-AN Quad from Teledyne DALSA". Note the bus and slot number of the installed board (this will be unique for each system unless systems are setup identically). Click on the **Diagnostic** button to view an analysis of the system PCI configuration space.



PCI Diagnostic 2.1

PCI device: X64-AN Quad from Teledyne DALSA (bus 5, slot 2, function 0)

Vendor ID 0x11EC
Device ID 0x2080
SubVendID 0x0001
SubsystID 0x3233
Rev. ID 0x00
IntLine 0x12
IntPin 0x01
Line size 0x00
Latency 0xFF
Min Grant 0xFF
Max Lat. 0xFF
Class Code 0x058000

Hex dump: ○ 8-bit ○ 16-bit ● 32-bit
addr 0

```
0x00> 0x208011EC
0x04> 0x02000106
0x08> 0x05800000
0x0C> 0x0000FF00
0x10> 0xF7800000
```

Command 0x0106
Status 0x0200
Header type 0x00
BIST 0x00

Base address registers

0 0xF7800000 to 0xF7BFFFFF Mem Pre 32-bit
1 ---- I/O Pre ---
2 ---- I/O Pre ---
3 ---- I/O Pre ---
4 ---- I/O Pre ---
5 ---- I/O Pre ---

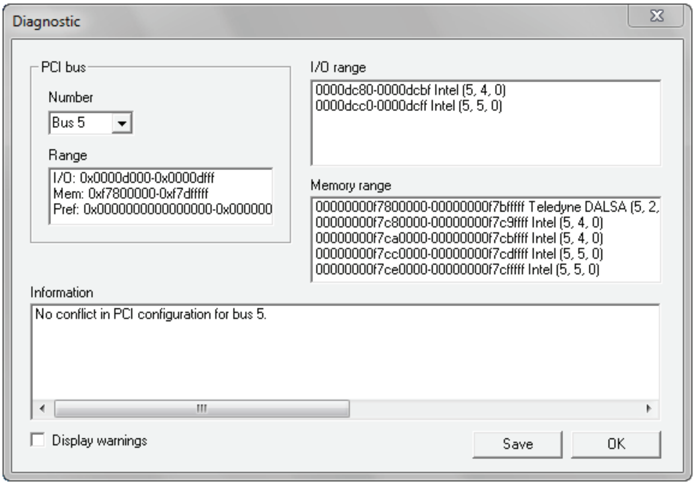
Expansion ROM ----

PCI-PCI bridge
Primary Bus ----
Second. Bus ----
Subord. Bus ----
Bridge Ctrl ----

Figure 10: Sapera Diagnostics Main Screen

Clicking on the **Diagnostic** button opens a new window with the diagnostic report. From the PCI Bus Number drop menu select the bus number where the X64-AN Quad is installed. In this example the slot is bus 5.

The window now shows the I/O and memory ranges used by each device on the selected PCI bus. The information display box will detail any PCI conflicts. If there is a problem, click on the **Save** button. A file named '**pcidiag.txt**' is created (in the …\Sapera LT\bin directory) with a full dump of the PCI configuration registers. Email this file when requested by the Technical Support group along with a full description of your computer.



```
I/O: 0x0000d000-0x0000dfff
Mem: 0xf7800000-0xf7dfffff
Pref: 0x0000000000000000-0x000000
```

```
0000dc80-0000dcbf Intel (5, 4, 0)
0000dcc0-0000dcff Intel (5, 5, 0)
```

```
00000000f7800000-00000000f7bfffff Teledyne DALSA (5, 2,
00000000f7c80000-00000000f7c9ffff Intel (5, 4, 0)
00000000f7ca0000-00000000f7cbffff Intel (5, 4, 0)
00000000f7cc0000-00000000f7cdffff Intel (5, 5, 0)
00000000f7ce0000-00000000f7cfffff Intel (5, 5, 0)
```

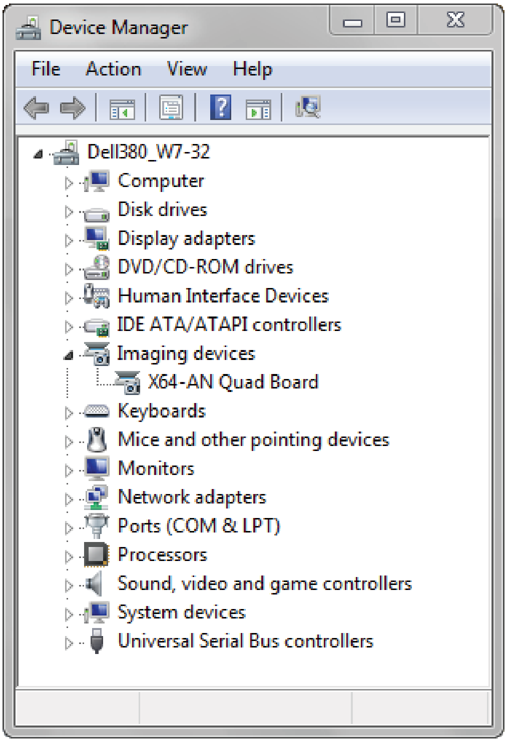
```
No conflict in PCI configuration for bus 5.
```

☐ Display warnings

Figure 11: Sapera Diagnostics-Reviewing PCI Bus Conflicts

Windows Device Manager

An alternative method to confirm the installation of the X64-AN Quad board and driver is to use the Windows Device manager tool. Use the Start Menu shortcut **Start • Settings • Control Panel • System • Hardware • Device Manager**. As shown in the following screen images, look for X64-AN Quad board under "Imaging Devices." Double-click and look at the device status. You should see "This device is working properly." Go to "Resources" tab and make certain that the device is mapped and has an interrupt assigned to it, without any conflicts.



```
Dell380_W7-32
    Computer
    Disk drives
    Display adapters
    DVD/CD-ROM drives
    Human Interface Devices
    IDE ATA/ATAPI controllers
    Imaging devices
        X64-AN Quad Board
    Keyboards
    Mice and other pointing devices
    Monitors
    Network adapters
    Ports (COM & LPT)
    Processors
    Sound, video and game controllers
    System devices
    Universal Serial Bus controllers
```

Figure 12: Windows Device Manager Screen

Sapera and Hardware Windows Drivers

Any problem seen after installation, such as an error message running CamExpert, first make certain the appropriate Teledyne DALSA drivers have started successfully during the boot sequence. Example, click on the **Start • Programs • Accessories • System Tools • System Information • Software Environment** and click on **System Drivers**. Make certain the following drivers have started for the X64-AN Quad driver.

Device	Description	Type	Started
CorX64An	X64-AN Quad	Kernel Driver	Yes
CorLog	Sapera Log viewer	Kernel Driver	Yes
CorMem	Sapera Memory manager	Kernel Driver	Yes
CorPci	Sapera PCI configuration	Kernel Driver	Yes
CorSerial	Sapera Serial Port manager	Kernel Driver	Yes

Table 1: X64-AN Quad Device Drivers

Teledyne DALSA Technical Support may request that you check the status of these drivers as part of the troubleshooting process.

Recovering from a Firmware Update Error

This procedure is required if any failure occurred while updating the X64-AN Quad firmware on installation or during a manual firmware upgrade. On the rare occasion the board has corrupted firmware, any Sapera application such as CamExpert or the grab demo program will not find an installed board to control.

Possible reasons for firmware loading errors or corruption are:

- Computer system mains power failure or deep brown-out.
- PCI bus or checksum errors.
- PCI bus timeout conditions due to other devices.
- User forcing a partial firmware upload using an invalid firmware source file.

When the X64-AN Quad firmware is corrupted, executing a manual firmware upload will not work because the firmware loader can not communicate with the board. In the extreme case, corrupted firmware may even prevent Windows from booting.

Solution**:** The user manually forces the board to initialize from protected firmware designed only to allow driver firmware uploads. When the firmware upload is complete, reboot the board to initialize it in its normal operational mode.

- This procedure requires removing the X64-AN Quad board several times from the computer.
- *Important*: Referring to the board's user manual (in the connectors and jumpers reference section), identify the configuration jumper location. The Boot Recovery Mode jumper for the X64-AN Quad is J15 (see "J15: Boot Recovery Mode" on page 70).
- Shut down Windows and power OFF the computer.
- Move the configuration switch for boot recovery (safe mode) from its default position to the boot recovery mode position.
- Power on the computer. Windows will boot normally.
- When Windows has started, do a manual firmware update procedure to update the firmware again (see "Executing the Firmware Loader from the Start Menu" on page 12).
- When the update is complete, shut down Windows and power off the computer.
- Set the Boot Recovery Mode switch back to its default position and reboot the computer once again.
- Verify that the frame grabber is functioning by running a Sapera application such as CamExpert.

Driver Information via the Device Manager Program

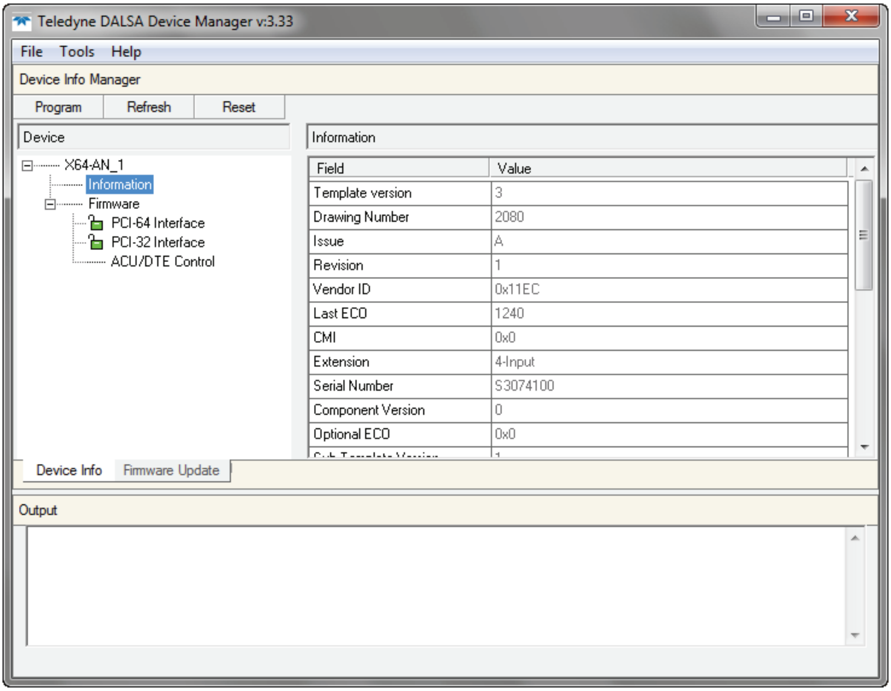
The Teledyne DALSA Device Manager program provides a convenient method of collecting information about the installed X64-AN Quad. System information such as operating system, computer CPU, system memory, PCI configuration space, plus X64-AN Quad calibration and firmware information can be displayed or written to a text file (default file name – BoardInfo.txt). Note that this program also manually uploads firmware to the X64-AN Quad (described elsewhere in this manual).

Execute the program via the Windows Start Menu shortcut **Start • All Programs • Teledyne DALSA • X64-AN Quad Device Driver • Device Manager**. If the Device Manager program does not run, it will exit with a message that the board was not found. Since the X64-AN Quad board must have been in the system to install the board driver, possible reasons for an error are:

- Board is not in the computer
- Board driver did not start or was terminated
- PCI conflict after some other device was installed

Information Window

The following figure shows the Device Manager information screen (with one X64-AN Quad installed in the system). Click to highlight one of the board components and the information for that item is shown on the right hand window, as described below.



Field	Value
Template version	3
Drawing Number	2080
Issue	A
Revision	1
Vendor ID	0x11EC
Last ECO	1240
CMI	0x0
Extension	4-Input
Serial Number	S3074100
Component Version	0
Optional ECO	0x0
Sub Template Version	1

Figure 13: Device Manager Information Window

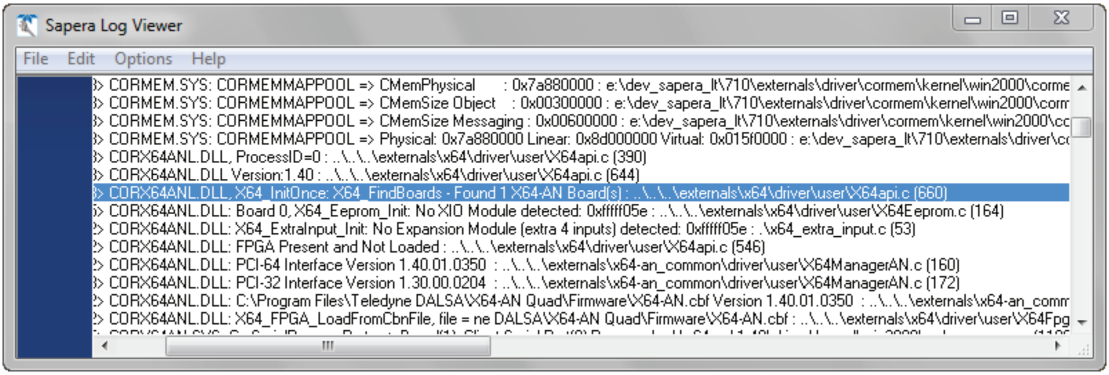
- Select Information to display identification and information stored in the X64-AN Quad firmware.
- Select Firmware to display version information for the firmware components.
- Select one of the PCI interface components to load custom firmware when supplied by Teledyne DALSA engineering for a future new feature.
- Click on File • Save Device Info to save all information to a text file. Default location is drive:\TeledyneDALSA\X64-AN Quad\Bin\BoardInfo.txt. Email this file when requested by Technical Support.

Teledyne DALSA Log Viewer

The third step in the verification process is to save in a text file the information collected by the Log Viewer program. Run the program via the Windows Start Menu shortcut **Start • All Programs • Teledyne DALSA • Sapera LT • Tools • Log Viewer**.

The Log Viewer lists information about the installed Teledyne DALSA drivers. Click on **File • Save** and you will be prompted for a text file name to save the Log Viewer contents. Email this text file to Technical Support when requested or as part of your initial contact email.

Although the information collected by the Log Viewer seems complicated, you can make some initial diagnostics by checking the status of the Teledyne DALSA driver. In the screen shot below, note the highlighted line which states [... CORX64ANL.DLL ... Found 1 *X64-AN* board (s) ...]. This confirms that the driver can communicate with the X64-AN Quad.



```
> CORMEM.SYS: CORMEMMAPPOOL => CMemPhysical     : 0x7a880000 : e:\dev_sapera_lt\710\externals\driver\cormem\kernel\win2000\corme
> CORMEM.SYS: CORMEMMAPPOOL => CMemSize Object   : 0x00300000 : e:\dev_sapera_lt\710\externals\driver\cormem\kernel\win2000\corm
> CORMEM.SYS: CORMEMMAPPOOL => CMemSize Messaging : 0x00600000 : e:\dev_sapera_lt\710\externals\driver\cormem\kernel\win2000\cc
> CORMEM.SYS: CORMEMMAPPOOL => Physical: 0x7a880000 Linear: 0x8d000000 Virtual: 0x015f0000 : e:\dev_sapera_lt\710\externals\driver\cc
> CORX64ANL.DLL, ProcessID=0 : ..\..\..\externals\x64\driver\user\X64api.c (390)
> CORX64ANL.DLL Version:1.40 : ..\..\..\externals\x64\driver\user\X64api.c (644)
> CORX64ANL.DLL, X64_InitOnce: X64_FindBoards - Found 1 X64-AN Board(s) : ..\..\..\externals\x64\driver\user\X64api.c (660)
> CORX64ANL.DLL: Board 0, X64_Eeprom_Init: No XIO Module detected: 0xfffff05e : ..\..\..\externals\x64\driver\user\X64Eeprom.c (164)
> CORX64ANL.DLL: X64_ExtraInput_Init: No Expansion Module (extra 4 inputs) detected: 0xfffff05e : .\x64_extra_input.c (53)
> CORX64ANL.DLL: FPGA Present and Not Loaded : ..\..\..\externals\x64\driver\user\X64api.c (546)
> CORX64ANL.DLL: PCI-64 Interface Version 1.40.01.0350  : ..\..\..\externals\x64-an_common\driver\user\X64ManagerAN.c (160)
> CORX64ANL.DLL: PCI-32 Interface Version 1.30.00.0204  : ..\..\..\externals\x64-an_common\driver\user\X64ManagerAN.c (172)
> CORX64ANL.DLL: C:\Program Files\Teledyne DALSA\X64-AN Quad\Firmware\X64-AN.cbf Version 1.40.01.0350  : ..\..\..\externals\x64-an_comm
> CORX64ANL.DLL: X64_FPGA_LoadFromCbnFile, file = ne DALSA\X64-AN Quad\Firmware\X64-AN.cbf : ..\..\..\externals\x64\driver\user\X64Fpg
```

Figure 14: Sapera Log Viewer Screen

Memory Requirements with Area Scan Acquisitions

The X64-AN Quad allocates by default two frame buffers in on-board memory, each equal in size to the acquisition frame buffer. This double buffering memory allocation is automatic at the driver level. Two buffers will ensure that the acquired video frame is complete and not corrupted in cases where the image transfer to host system memory may be interrupted and delayed by other host system processes.

That is, the image acquisition to one frame buffer is not interrupted by any delays in transfer of the other frame buffer (which contains the previously acquired video frame) to system memory. Note that an application can change the number of on-board frame buffers using the Sapera LT API.

If allocation for the requested number of buffers fails, the driver will reduce the number of on-board frame buffers requested until they can all fit. When reaching 2 on-board buffers, if they still cannot fit, the driver will reduce the size such that it allocates two partial buffers. This mode will write image data to the buffer while wrapping image lines around to the beginning of a buffer when full. This mode relies on reading out the image data to the host computer faster than the acquisition.

Symptoms: CamExpert Detects no Boards

- **If using Sapera version 5.20 or later:**

When starting CamExpert, if no Teledyne DALSA board is detected, CamExpert will start in offline mode. There is no error message and CamExpert is functional for creating or modifying a camera configuration file. If CamExpert should have detected the installed board, troubleshoot the installation problem as described below.

Troubleshooting Procedure

When CamExpert detects no installed Teledyne DALSA board, there could be a hardware problem, a PnP problem, a PCI problem, a kernel driver problem, or a software installation problem.

- Make certain that the card is properly seated in the PCI slot.
- Perform all installation checks described in this section ("Troubleshooting Problems" on page 23) before contacting Technical Support.
- Try the board in a different PCI slot, if the board does not initialize.

Symptoms: X64-AN Quad Does Not Grab

You are able to start Sapera CamExpert but you do not see an image and the frame rate displayed is 0.

- If your camera is powered through a camera cable, make certain that J17 on the X64-AN Quad is connected to a floppy power cable. Otherwise, the camera must be powered using an external power supply.
- Verify the camera and timing parameters with the camera in free run mode.
- Make certain that you provide an external trigger if the camera configuration file requires one. Use the software trigger feature of CamExpert if you do not have a trigger source.
- Does your camera provide a WEN signal that you need to use?
Adapt your configuration file and camera cable accordingly.
- Make certain that the pinout of your camera cable matches your camera and that the camera is properly connected to the cable.
- Make certain that the camera is configured for the proper mode of operation (Composite Video, Separate Sync, Master Mode). This must match the camera configuration file. Refer to your camera datasheet.

- Try using a standard video source (RS-170 or CCIR). This validates that X64-AN Quad is able to grab and may point to a problem with your camera configuration file.
- Try to snap one frame instead of continuous grab.
- Perform all installation checks described in this section ("Troubleshooting Problems" on page 23) before contacting Technical Support.

Symptoms: Card grabs black

You are able to use Sapera CamExpert, the displayed frame rate is as expected, but the display stays black.

- Set your camera to manual exposure mode and set the exposure to a longer period plus open the lens iris.
- Try changing Contrast/Brightness settings.
- Try changing the clamping setting (DC restoration) if it is not a standard video source configuration file and the camera file was not supplied by Teledyne DALSA.
- Try using a standard video source (RS-170 or CCIR). This validates that X64-AN Quad is able to grab and may point to a problem with your camera configuration file.
- Try to snap one frame instead of continuous grab.
- Make certain that the input LUT is not programmed to output all '0's.
- This problem is sometimes caused by a PCI transfer issue. No PCI transfer takes place, so the frame rate is above 0 but nevertheless no image is displayed in CamExpert.
- Make certain that BUS MASTER bit in the PCI configuration space is activated. Look in PCI Diagnostics for **BM** button under "Command" group. Make certain that the **BM** button is activated.
- Perform all installation checks described in this section ("Troubleshooting Problems" on page 23) before contacting Technical Support.

Symptoms: Card acquisition bandwidth is less than expected

The X64-AN Quad acquisition bandwidth is less than expected.

- Review the system for problems or conflicts with other expansion boards or drivers.
- Remove other PCI Express, PCI-32 or PCI-64 boards and check acquisition bandwidth again. Engineering has seen this case where other PCI boards in some systems cause limitations in transfers. Each system, with its combination of system motherboard and PCI boards, will be unique and must be tested for bandwidth limitations affecting the imaging application.

Theory of Operation

Camera Control and Synchronization

Source of Synchronization

The X64-AN Quad can use a variety of synchronization sources allowing it to interface with various cameras. All four inputs can make use of their own horizontal sync (HS), vertical sync (VS), frame reset and WEN signals. The Acquisition and Control Unit (ACU) is the main controller responsible for supervising the acquisition process. It manages all the signals coming from the cameras and recovers the timing information to accurately digitize the video signal into pixels.

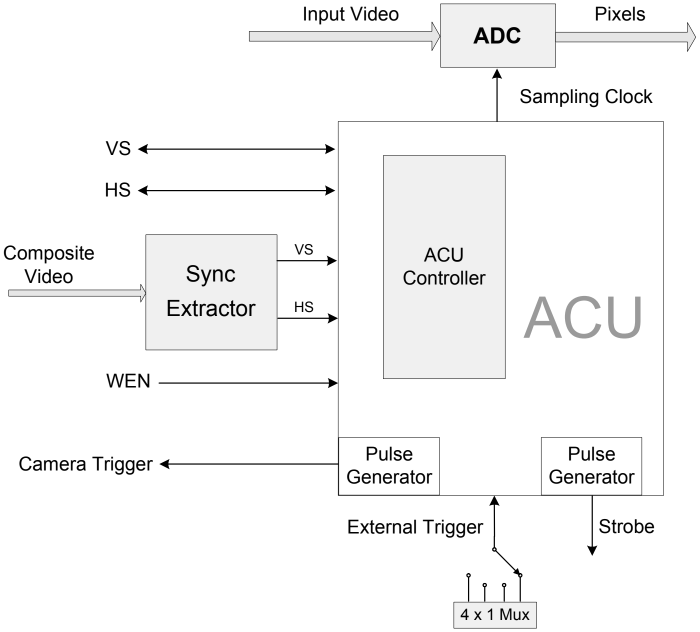


Figure 15: Synchronization Block Diagram: 1 Input Shown

Additional details on the various synchronization modes follow.

Sync on Composite Video

Vertical Sync (VS) and horizontal sync (HS) signals are extracted from the composite video output signal by the sync extractor The PLL receives the stripped horizontal sync and outputs a pixel clock which is line-locked to the incoming video. The pixel clock then drives the ADC timing to digitize video and also generates frame timing. The PLL, which maintains a frequency stable pixel clock, is programmed based on the timing requirements of the incoming video.

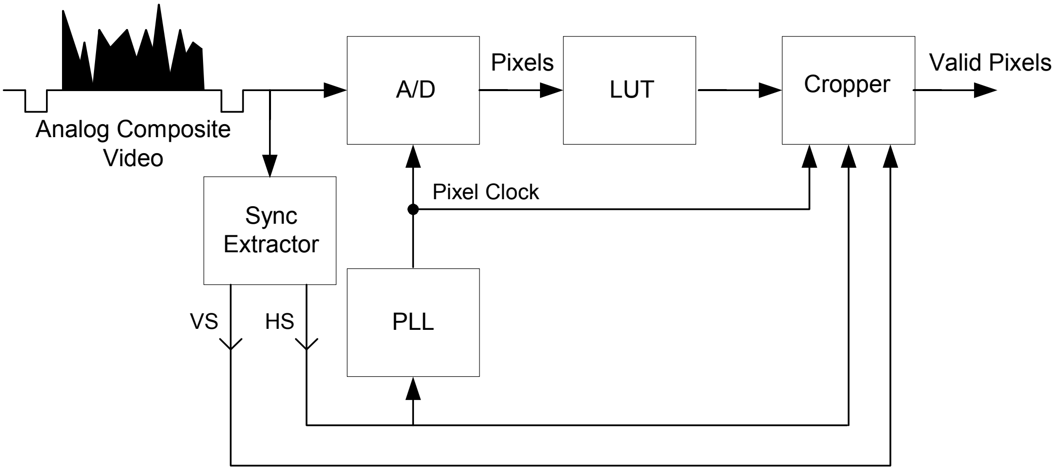


Figure 16: Composite Video Synchronization Block Diagram

Synchronization on composite video is commonly used for standard RS-170 and CCIR cameras as well as for many non-standard cameras. The following table shows the Sapera timing parameters used by the synchronization hardware. Use the Sapera CamExpert utility to program camera timings and create camera files for any non-standard camera usable with the X64-AN Quad.

Sapera parameters for Sync on Composite Video:

CORACQ_PRM_SYNC = CORACQ_VAL_SYNC_COMP_VIDEO

CORACQ_PRM_HSYNC: Size of horizontal sync pulse

CORACQ_PRM_HBACK_PORCH: Size of horizontal back porch

CORACQ_PRM_HACTIVE: Number of valid pixels per line

CORACQ_PRM_HFRONT_PORCH: Size of horizontal front porch

CORACQ_PRM_VSYNC: Size of vertical sync pulse

CORACQ_PRM_VBACK_PORCH: Size of vertical back porch

CORACQ_PRM_VACTIVE: Number of valid line from camera

CORACQ_PRM_VFRONT_PORCH: Size of vertical front porch

Sync on Separate Sync

In this mode the VS and HS signals are each input to the X64-AN Quad. The sync extractor circuit is not used. The PLL compares the separate horizontal sync input to the internal feedback and generates the PLL clock. The ADC uses the PLL clock to digitize the video input. The polarity of the sync inputs can be negative or positive. The incoming signals must be referenced to system ground.

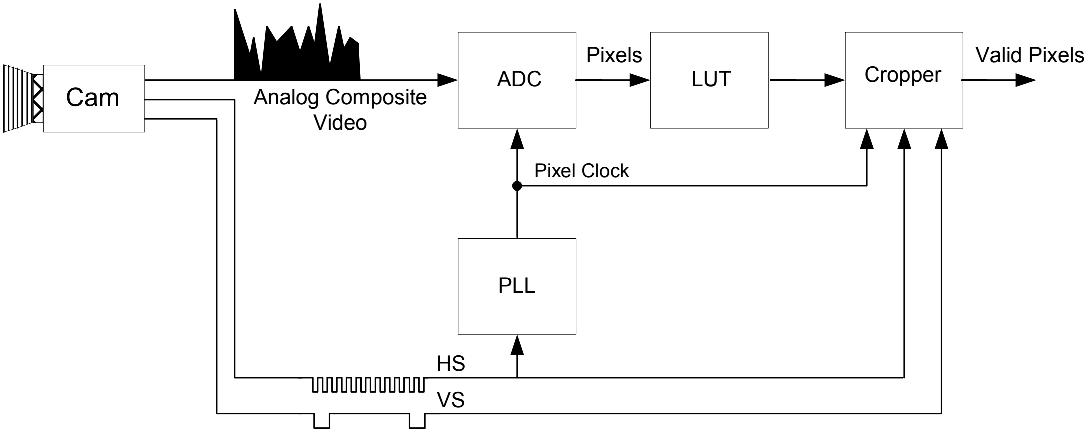


Figure 17: Separate Sync Video Synchronization Block Diagram

Each camera has its own VS and HS output which are typically TTL level.

Sapera parameters for Sync on Separate Sync:

CORACQ_PRM_SYNC = CORACQ_VAL_SYNC_SEP_SYNC

CORACQ_PRM_HSYNC: Size of horizontal sync pulse

CORACQ_PRM_HSYNC_POLARITY = CORACQ_VAL_ACTIVE_LOW or CORACQ_VAL_ACTIVE_HIGH

CORACQ_PRM_HBACK_PORCH: Size of horizontal back porch

CORACQ_PRM_HACTIVE: Number of valid pixels per line

CORACQ_PRM_HFRONT_PORCH: Size of horizontal front porch

CORACQ_PRM_VSYNC: Size of vertical sync pulse

CORACQ_PRM_VSYNC_POLARITY = CORACQ_VAL_ACTIVE_LOW or CORACQ_VAL_ACTIVE_HIGH

CORACQ_PRM_VBACK_PORCH: Size of vertical back porch

CORACQ_PRM_VACTIVE: Number of valid line from camera

CORACQ_PRM_VFRONT_PORCH: Size of vertical front porch

Internal Sync

In Internal Sync mode, a clock generator is programmed to generate the desired pixel clock and time base signals. The clock generator produces separate horizontal and vertical sync signals that match the desired video format. These signals, called horizontal drive (HD) and vertical drive (VD), are then output to the camera. The frequency synthesizer can be programmed to generate any clock frequency up to 50MHz with less than 1ns jitter. Internal Sync mode easily supports both standard and non-standard camera timing.

The horizontal and vertical timing created by the clock generator is output to the camera as HD/VD signals: called Master Mode. It is also possible to deactivate the VD output for use with cameras that have such a requirement.

Note that the X64-AN Quad in Master Mode is independent from the selected source of synchronization. It is therefore possible for X64-AN Quad to send VD/HD to the camera, but still synchronize to the VS/HS present in the composite video signal.

X64-AN Quad allows the same VD/HD to be sent to all four cameras simultaneously. This is useful to genlock cameras together. X64-AN Quad has two master mode controllers allowing control of up to two independent sets of cameras.

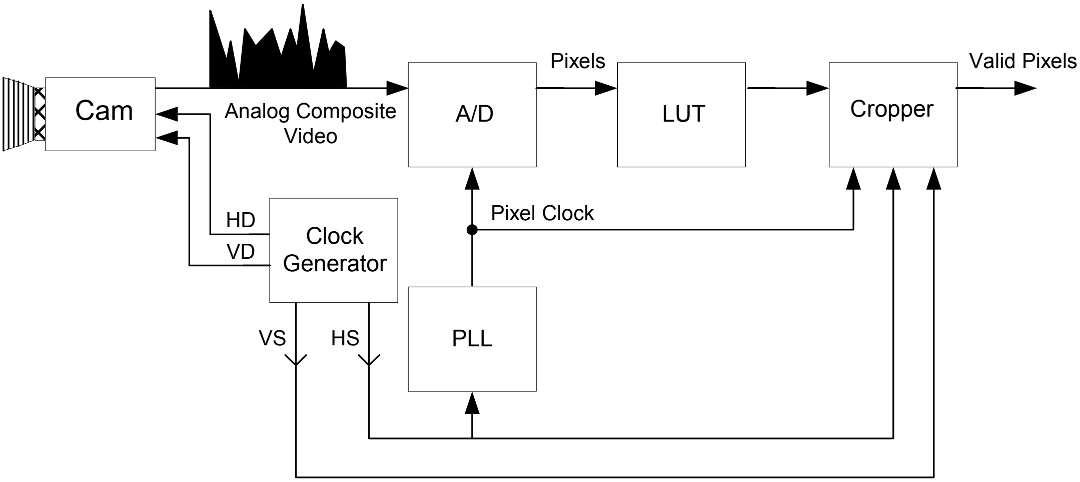


Figure 18: Internal Sync in Master Mode Block Diagram

When using Internal Sync, the horizontal reference for acquisition is HD. This is equivalent to a horizontal front porch of 0 pixels. The horizontal signal is used as a time reference to configure the clamping pulse delay and duration parameters.

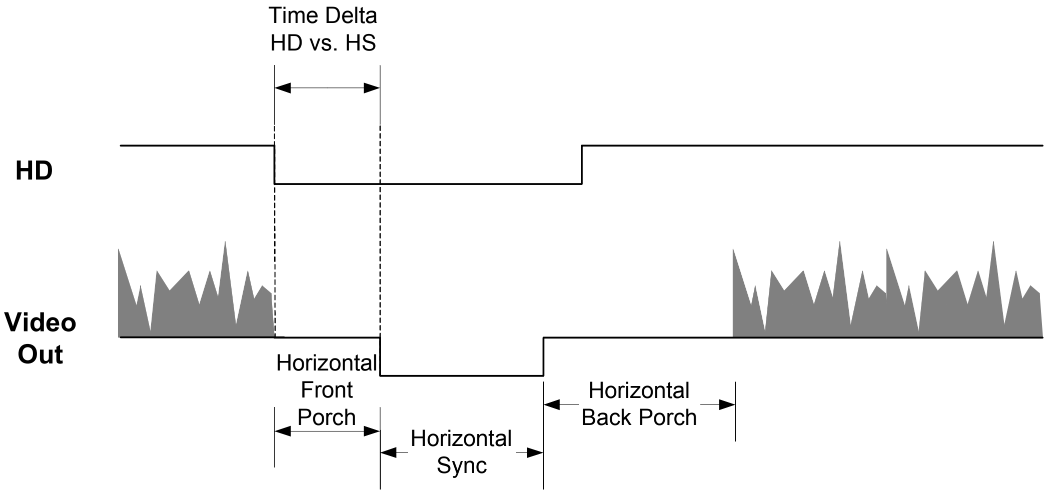


Figure 19: Master Mode Timing - HD relation to HS

Sapera parameters for Sync on Internal Sync:

CORACQ_PRM_SYNC = CORACQ_VAL_SYNC_INT_SYNC

CORACQ_PRM_MASTER_MODE =
CORACQ_VAL_MASTER_MODE_HSYNC_VSYNC

CORACQ_PRM_MASTER_MODE_HSYNC_POLARITY = {
CORACQ_VAL_ACTIVE_LOW, CORACQ_VAL_ACTIVE_HIGH}

CORACQ_PRM_MASTER_MODE_VSYNC_POLARITY = {
CORACQ_VAL_ACTIVE_LOW, CORACQ_VAL_ACTIVE_HIGH}

CORACQ_PRM_HSYNC: Size of horizontal sync pulse

CORACQ_PRM_HBACK_PORCH: Size of horizontal back porch

CORACQ_PRM_HACTIVE: Number of valid pixels per line

CORACQ_PRM_HFRONT_PORCH: Size of horizontal front porch

CORACQ_PRM_VSYNC: Size of vertical sync pulse

CORACQ_PRM_VBACK_PORCH: Size of vertical back porch

CORACQ_PRM_VACTIVE: Number of valid line from camera

CORACQ_PRM_VFRONT_PORCH: Size of vertical front porch

WEN

Some cameras indicate when valid data is output by generating a write enable signal (WEN). The function of WEN is similar to a vertical sync pulse. When enabled, the X64-AN Quad uses WEN as the vertical timing reference instead of VS. Some cameras generate WEN, but with no VS pulse embedded in the composite video signal.

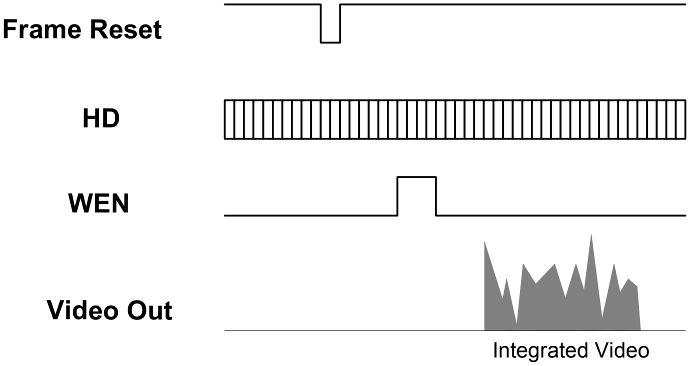


Figure 20: WEN Timing

One WEN signal is available per connected camera. WEN has programmable polarity; it can be selected to be active high or active low.

Sapera parameters for WEN :

CORACQ_PRM_WEN_ENABLE = {TRUE, FALSE}

CORACQ_PRM_WEN_POLARITY = { CORACQ_VAL_ACTIVE_LOW, CORACQ_VAL_ACTIVE_HIGH}

CORACQ_PRM_VBACK_INVALID: Number of lines to skip for valid video after WEN pulse

Camera Control

Pulse Generator

X64-AN Quad has three independent timers per input that control pulse generation for camera controls. This allows to position pulses precisely (to a resolution of 1µs) relative to the trigger event. Pulse generation flexibility is required to support a wide range of camera control modes (edge pre-select, pulse width control, E-Donpisha, etc.).

An independent timer is available for each of the following signals:

- VS
- Frame Reset
- Strobe

VS and Frame Reset timers can be combined to generate a double-pulse on the same camera signal control pin. This is required for some camera modes, like long time exposure.

Each timer has the following capabilities:

- Programmable polarity (active high or active low)
- Programmable delay from trigger event (up to 65 seconds)
- Programmable duration (up to 65 seconds)

Timer granularity (timer step size) is 1µs when the delay and duration values are below 65ms. Granularity reduces to 1ms for a delay or duration above 65ms. Delay and duration always have the same granularity level. Therefore even if only one timer has the reduced granularity of 1ms, all timers then have the same 1ms step size.

Each timer can be started by any of the following events:

- VS (default)
- External trigger
- Internal trigger
- Software trigger

Frame Reset

Frame reset—also known as camera reset or camera trigger—is a signal sent by the X64-AN Quad to the camera which triggers an acquisition. One frame reset signal is available per X64-AN Quad input. The pulse duration and polarity are programmable. Frame reset can be triggered either by an external trigger signal, an internal trigger, a software trigger or a VS event. After the trigger is initiated, an internal frame reset counter counts up to a maximum of 65 seconds. This pulse is normally used to control the exposure of the camera CCD (used with camera modes such as Edge Pre-Select or Pulse Width Control).

Two parameters control the frame reset pulse. First, the offset parameter sets the delay from trigger before asserting frame reset. Then, the size parameter specifies the frame reset pulse duration, which in turn, controls the exposure period on some cameras.

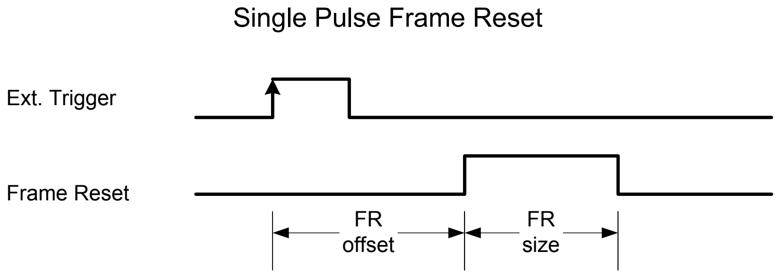


Figure 21: Single pulse frame reset timing

Sapera parameters for Frame Reset :

Refer to Time Integration method of Sapera documentation.

CORACQ_CAP_TIME_INTEGRATE_METHOD:
Method to use for time integration

CORACQ_PRM_TIME_INTEGRATE_PULSE0_DELAY:
Pulse offset from trigger event

CORACQ_PRM_TIME_INTEGRATE_PULSE0_DURATION:
Size of pulse

CORACQ_PRM_TIME_INTEGRATE_PULSE0_POLARITY =
{ CORACQ_VAL_ACTIVE_LOW, CORACQ_VAL_ACTIVE_HIGH}

VSYNC

One VS signal is output per active camera. This feature is used on some cameras to control the exposure rate when in Master Mode. The VS can generate up to two pulses, each with a different duration up to 65 seconds.

Two parameters control the VS pulse. First, the offset sets the delay from trigger before asserting VS. Then, the duration sets the VS pulse width.

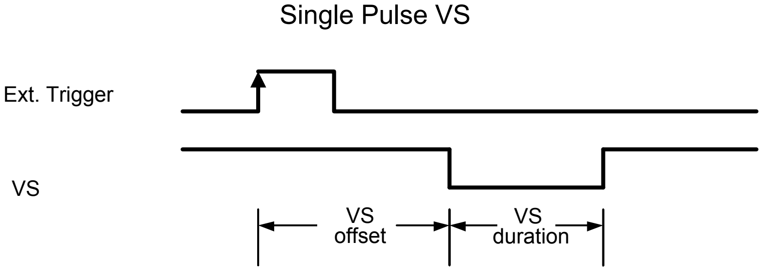


Figure 22: Single pulse VS timing

Sapera parameters for VSync :

Refer to Time Integration method of Sapera documentation

CORACQ_CAP_TIME_INTEGRATE_METHOD:
Method to use for time integration

CORACQ_PRM_TIME_INTEGRATE_PULSE_DELAY:
Pulse offset from trigger event

CORACQ_PRM_TIME_INTEGRATE_PULSE_DURATION:
Size of pulse

CORACQ_PRM_TIME_INTEGRATE_PULSE_POLARITY =
{ CORACQ_VAL_ACTIVE_LOW, CORACQ_VAL_ACTIVE_HIGH}

External Trigger and Strobe

External Trigger

External Trigger allows image acquisitions to be synchronized to external events. With this mode enabled, when the X64-AN Quad receives a trigger signal, the acquisition begins with the next valid frame.

One external trigger signal is available per input. The same external trigger signal can be used to synchronized acquisitions from multiple inputs. The external trigger input uses an opto-coupler for isolation and protection. The trigger source must drive at least 2mA to turn on the opto-coupler. Minimum voltage difference is 2V to turn on and 0.8V to turn off. The external trigger can be either rising or falling edge. Maximum input differential voltage supported by opto-coupler is 24V.

The incoming trigger pulse is “debounced” to ensure that no voltage glitch would be detected as a valid trigger pulse. This debounce circuit time constant can be programmed from 1μs to 255μs. Any pulse smaller than the programmed value is blocked and therefore not seen by the acquisition circuitry.

Note: If no debounce value is specified (value of 0μs), the minimum value of 1μs will be used.

Figure 23: External Trigger Input

Let	t(et) = time of external trigger in µs
	t(vt) = time of validated trigger in µs
	t(oc) = time opto-coupler takes to change state
	t(d) = debouncing duration from 1 to 255µs
trigger high	For an active high external trigger, t(oc) = 10µs:
	t(vt) = t(et) + 10µs + t(d)
trigger low	For an active low external trigger, t(oc) = 50µs:
	t(vt) = t(et) + 50µs + t(d)

Note: Teledyne DALSA recommends using an active high external trigger to minimize the time it takes for the opto-coupler to change state. Specifically, the opto-coupler response time is 10µs for active high compared to 50µs for active low.

If the duration of the external trigger is > t(oc) + t(d), then a valid acquisition trigger is detected. Therefore, the external pulse with active high polarity must be at least 11µs (if debounce time is set to 1) in order to be acknowledged. Any pulse larger than 51µs is always considered valid.

It is possible to emulate an external trigger using the software trigger which is generated by a function call from an application.

External trigger input is available on J5, a DB9 connector (see "J5 – Trigger Signals Connector" on page 65). The X64-AN Quad external trigger interfaces to the external world through the use of an opto-coupled device. Formed by a LED emitter combined with a photo-detector in close proximity, an opto-coupler (or opto-isolator) connects the X64-AN Quad external trigger input and the user circuit together while using separate grounds. This "galvanic isolation" approach prevents ground loops and protects both circuits. A serial resistor limits the current.

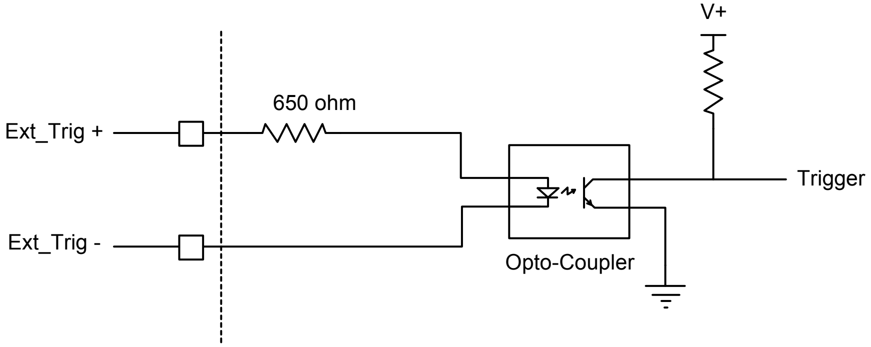


Figure 24: Opto-coupler Detail

The surrounding circuit that converts the voltage to flow as a current into the LED is therefore crucial to the good performance of the opto-coupler. The emitted light will not turn the transistor ON if the current flowing through the LED is too small.

Electrical parameters	**Description**	**Value**
$V_{IL\ max}$	Maximum voltage different to turn OFF	0.8 V
$V_{IH\ min}$	Minimum voltage difference to turn ON	2 V
$I_{I\ min}$	Minimum input current to turn ON	2 mA
$I_{I\ max}$	Maximum input current to turn ON	50 mA
$t_{I\ min}$	Minimum input pulse width to turn ON	10 us
$V_{f\ max}$	Maximum forward voltage	24 V
$V_{r\ max}$	Maximum reverse voltage	-25 V

Note: TTL signals are approximately 0 and 5V, corresponding to logical 0 and 1, respectively. A standard TTL output can sink 16mA and could be used as a sink to drive an opto-coupled input. That is, +5V is connected to Ext_Trig+ and the sink trigger source is connected to Ext_Trig-.

Some logic devices will not supply enough current to reliably drive the Ext_Trig+ of an opto-coupled input: a buffer between that logic device output and the Ext_Trig+ input is required. One possibility is a CMOS buffer with TTL compatible inputs, such as the 74AC240 (inverting buffer) or 74AC241 (non-inverting buffer). These devices can supply up to 24mA, close to the supply voltage.

Sapera parameters for External Trigger:

CORACQ_PRM_EXT_TRIGGER_ENABLE =
CORACQ_VAL_EXT_TRIGGER_ON

CORACQ_PRM_EXT_TRIGGER_SOURCE =
{0: Same trigger source number as video source, 1: Trigger source 1, 2: Trigger source 2, 3: Trigger source 3, 4: Trigger source 4}

CORACQ_PRM_EXT_TRIGGER_DETECTION =
{CORACQ_VAL_RISING_EDGE, CORACQ_VAL_FALLING_EDGE,
CORACQ_VAL_ACTIVE_LOW. CORACQ_VAL_ACTIVE_HIGH}

CORACQ_PRM_EXT_TRIGGER_DURATION: Debouncing duration

CORACQ_PRM_EXT_TRIGGER_FRAME_COUNT: Number of frame to acquire per trigger

Note: CORACQ_PRM_EXT_TRIGGER_LEVEL always represents the opto-coupler trigger input independent of its actual value (CORACQ_VAL_LEVEL_TTL or CORACQ_VAL_LEVEL_422). This means this parameter "does not matter" for the X64-AN Quad driver.

Strobe

One strobe signal is available per input. See section "J19: Strobe & Com Ports" on page 69 for pinout information. The polarity and pulse duration are programmable (up to 65 seconds). The strobe signal is achieved using a 74AHCT125 driver with the following electrical characteristics:

Electrical parameters	Description	Value
$V_{OH\ typ}$	Typical high-level output voltage	3.9V
$I_{OH\ max}$	Maximum high-level output current	-8mA (sourcing)
$I_{OL\ max}$	Maximum low-level output current	8mA (sinking)

Sapera parameters for Strobe :

Refer to Strobe Method in Sapera documentation

CORACQ_PRM_STROBE_ENABLE = TRUE

CORACQ_PRM_STROBE_METHOD =
{CORACQ_VAL_STROBE_METHOD_1, CORACQ_VAL_STROBE_METHOD_2,
CORACQ_VAL_STROBE_METHOD_4}

CORACQ_PRM_STROBE_POLARITY =
{CORACQ_VAL_ACTIVE_LOW, CORACQ_VAL_ACTIVE_HIGH}

CORACQ_PRM_STROBE_DELAY: Pulse offset from trigger event

CORACQ_PRM_STROBE_DELAY_2: Duration of exclusion region

CORACQ_PRM_STROBE_DURATION: Pulse duration

Serial Port

X64-AN Quad hosts 4 serial ports intended for camera control only. Due to data flow multiplexing of the 4 serial ports, only one serial port can communicate at one time. Data received from a camera will always be sent to the last serial port that sent characters. See section "J19: Strobe & Com Ports" on page 69 for pinout information.

The default names for the serial ports are: X64-AN Quad***_X_*****Serial*****_Y***, where *X* represents the X64-AN Quad board number from 1 to 8, and *Y* represents the serial port number from 0 to 3.

Note: A typical configuration would use 9600 baud♦8-bit♦no parity♦1 stop bit (9600-8-N-1).

Ports can be used with their default names (for example: X64-AN Quad_1_Serial_0) by many camera control applications. Additionally, the serial port can be mapped as a standard Windows COMx port for convenience or compatibility with any communication program (such as HyperTerminal).

Acquisition Process

The following sections describe the various acquisition stages of the X64-AN Quad. Composite analog video input from cameras or any other source can be processed in both the analog domain before the A/D stage and in the digital domain before transfer to host system frame buffers.

Anti-aliasing Filter

Following a differential input buffer stage, the video passes through a low-pass filter, optimized for standard video frequencies with a filter corner set to 12.87 MHz. When acquiring video from non-standard sources, the low-pass filter can be bypassed with a manually set jumper (see "J8, J9, J10, J11: Input Low Pass Filter Select" on page 65). Each of the four inputs has its own low-pass filter bypass jumper.

The low-pass filter strips high frequency signal content from the incoming video signal, to avoid sampling aliasing artifacts. Standard video (RS-170, CCIR) has useful frequency content up to approximately 6MHz. Frequencies above this can be eliminated using the low-pass filter. Sampling rates for standard video are 10MHz to 14MHz. If frequencies at or above the sampling rate are present in the input, they represent noise rather than useful video. These frequencies can "alias" into the real video signals causing corruption. The low-pass filter eliminates any high frequency signal content before digitization. See "Input Block Diagram (one shown)" on page 5.

Contrast and Brightness Adjustment

Contrast and Brightness are controlled through the input gain of the ADC. The X64-AN Quad is calibrated for standard video during manufacturing. The figure below shows the relationship between input analog video (vertical axis) relative to the output digital data (horizontal axis).

Brightness controls the offset of the digitization line while contrast controls its gain (the slope of the line). The vertical axis represents the voltage level of the incoming video signal (black level is 53mV, white level is 714mV for RS170 video) while the horizontal axis shows the resulting pixel value. As an example, using default brightness and contrast settings, an incoming video signal of 350mV will be digitized to a value of 115. By increasing the brightness, the digitization line gets shifted down (with no slope change). This creates a brighter digital image (same input voltage leads to a higher pixel value). Likewise a decrease in brightness leads to a darker image (same input voltage leads to a lower pixel value). Changing the brightness only affects the offset of the digitization line, not its slope.

Contrast controls the relationship between a change in input analog video to the change in digital pixel value. Increasing contrast causes a greater difference in the digital pixel values for any constant change in input video level. Likewise a decrease in contrast reduces the difference in the digital pixel values for any constant change in input video level.

Default values of brightness and contrast use the full resolution of the X64-AN Quad ADC based on the Sapera acquisition parameters Video Level Min/Max.

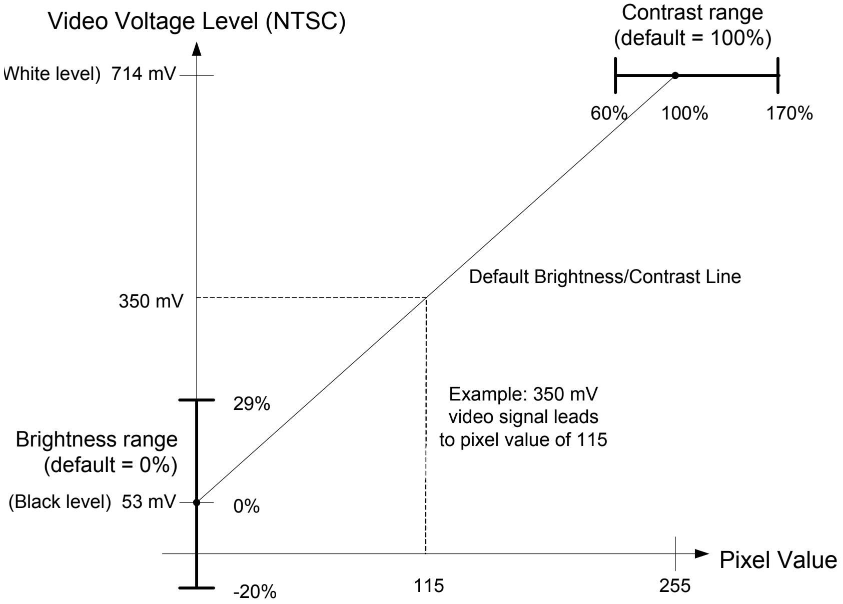


Figure 25: Brightness & Contrast Relationship

Sapera parameters for Contrast and Brightness :

CORACQ_PRM_CONTRAST: Contrast percentage

CORACQ_PRM_BRIGHTNESS: Brightness percentage

Note: For the X64-AN Quad under Sapera, contrast percentage ranges from 60% to 170% with 100% being the default value. Brightness percentage ranges from –20% to 29% with 0 % being the default value.

A/D Converter

The X64-AN Quad uses a high speed Analog to Digital Converter (ADC). The ADC outputs a 8-bit unsigned binary values from 0x0 to 0xFF based on: the sampled analog input signal level, the ADC sampling window, and by the clamping voltage level. The Pixel Clock used by the ADC to sample the analog video comes from the PLL clock.

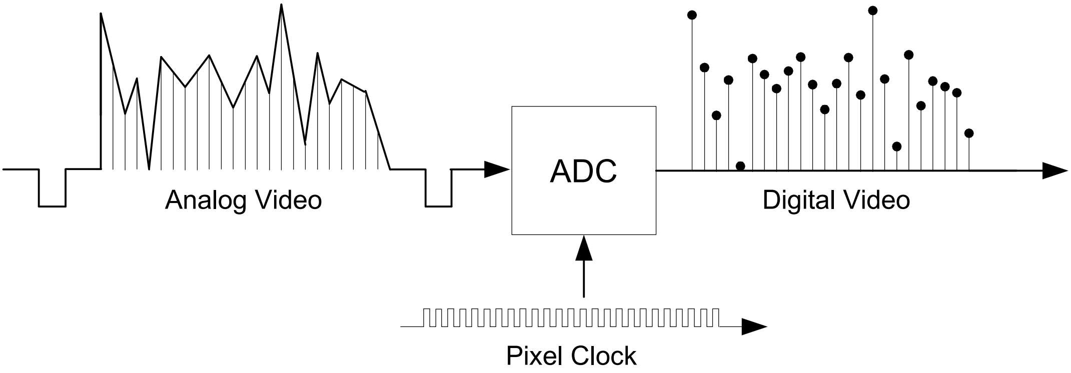


Figure 26: A/D Converter diagram

DC Restoration

DC Restoration uses a programmable clamp pulse. The horizontal back porch is used to establish the reference video black level.

Two parameters are required to locate the region used as the reference. The *clamp_start* and *clamp_end* parameters are referenced to the start of HS. The *clamp_width* is the time difference between *clamp_start* and *clamp_end*. See below for diagram.

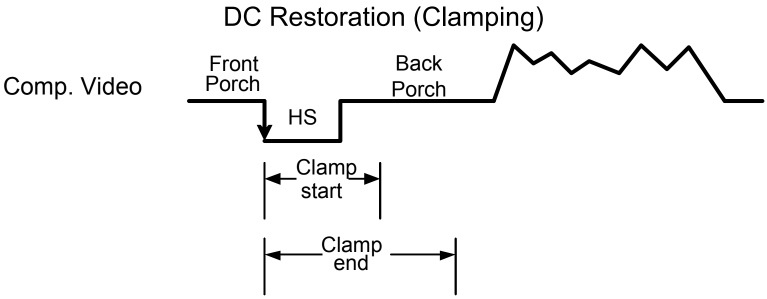


Figure 27: DC restoration timing

Note: With Internal Sync mode, the horizontal reference is the edge of HD. With this synchronization scheme, clamp start and clamp end values refer to first edge of HD, not to HS coming from composite video.

Sapera parameters for ADC Converter:

CORACQ_PRM_PIXEL_CLK_INT: Internal pixel clock frequency (in Hz)

CORACQ_PRM_PIXEL_DEPTH = 8

CORACQ_PRM_DC_REST_MODE =
{CORACQ_VAL_DC_REST_MODE_AUTO,
CORACQ_VAL_DC_REST_MODE_ON, CORACQ_VAL_DC_REST_MODE_OFF}

CORACQ_PRM_DC_REST_START: Start of clamp pulse relative to HS or HD

CORACQ_PRM_DC_REST_WIDTH: Clamp pulse duration

Lookup Table

Each of the 4 video inputs has an independent Lookup Table (LUT). The LUT format is 8-bits in and 8-bits out. The LUT is used for operations such as gamma adjustments, invert and threshold processes, etc.

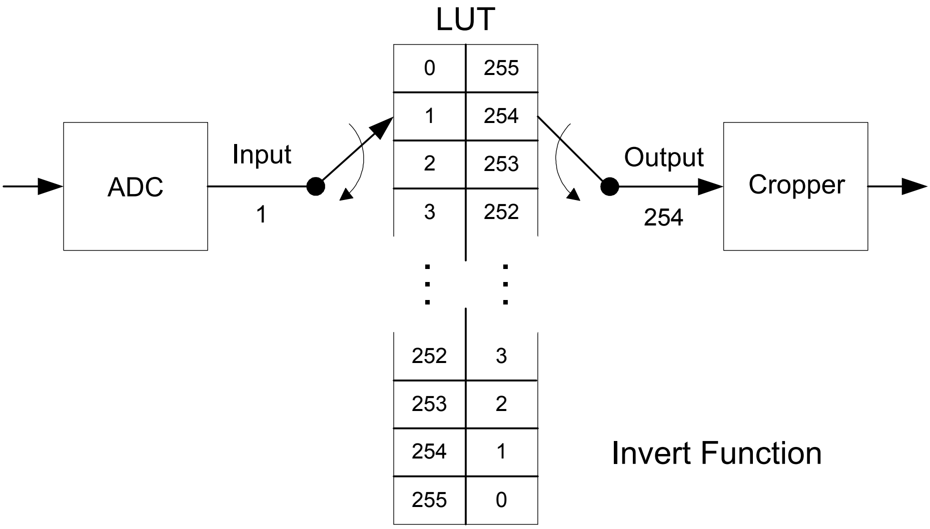


Figure 28: Lookup Table Example

Sapera parameters for Lookup Table:

CORACQ_PRM_LUT_ENABLE = {TRUE, FALSE}

Use CorAcqSetLut() to load a LUT into X64-AN Quad.

Cropper

The Cropper extracts a window from the incoming image. This window is represented by a rectangle where the upper-left corner is given by horizontal and vertical offset from the start of valid video and the rectangle size by width and height parameters. See below for diagram. Note that image widths must be a multiple of 16 bytes. For interlace scan video, image heights must be a multiple of two lines.

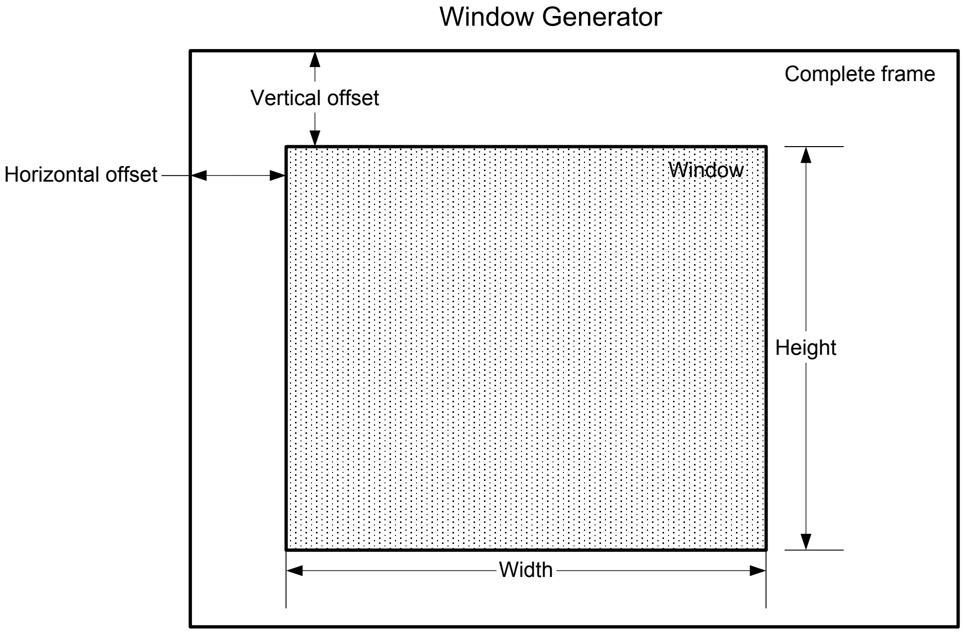


Figure 29: Cropper window example

Partial Scan Mode

Partial Scan mode reduces the number of lines output by a camera in order to increase the frame rate. X64-AN Quad sees the output of a partial scan camera as an image with reduced height. One simply needs to adjust the frame rate (vertical sync frequency) and image height in order to acquire from a partial scan mode camera.

Sapera Parameters for Cropper:

CORACQ_PRM_CROP_LEFT: Horizontal offset

CORACQ_PRM_CROP_HEIGHT: Vertical height of area of interest

CORACQ_PRM_CROP_TOP: Vertical offset

CORACQ_PRM_CROP_WIDTH: Horizontal width of area of interest

On-board Memory

The onboard memory behaves as a temporary buffer between the camera interface and the host PCI-bus system. The default total onboard memory capacity is 128MB. There is a maximum frame size of 4094 x 16,777,215. Two or more frames are stored within onboard memory for double buffering. Onboard memory allows the capture from cameras requiring a bandwidth exceeding the PCI-32 maximum of 132MB/second (PCI-64 maximum burst transfer rates are greater).

The X64-AN Quad supports the pixel format of 8-bit monochrome.

PCI Bus DMA Controller

The PCI Bus DMA controller has scatter/gather support to reduce CPU usage to a minimum. Host system memory allocated for frame buffers is virtually contiguous but physically scattered throughout all available memory. The buffer descriptor list is maintained in host memory.

The PCI Bus DMA controller maximum performance is specified independently from the analog acquisition front end of the X64-AN Quad. The following table defines the PCI Bus DMA controller maximum data transfer rates

Bus	Sustained Transfer	Max. Burst Transfer
PCI-32	120 MB/second	132 MB/second
PCI-64	320 MB/second	528 MB/second

Trigger to Image Reliability

Trigger-to-image reliability incorporates all stages of image acquisition inside an integrated controller to increase reliability and simplify error recovery. The trigger-to-image reliability model brings together all the requirements for image acquisition to a central management unit. These include signals to control camera timing, on-board frame buffer memory to compensate for PCI bus latency, and comprehensive error notification. If the X64-AN Quad detects a problem, the application can take appropriate action to return to normal operation.

The X64-AN Quad is designed with a robust ACU (Acquisition and Control Unit). The ACU monitors in real-time, the acquisition state of the input plus the DTE (Data Transfer Engine) which transfers image data from on-board memory into PC memory. In general, these management processes are transparent to end-user applications. With the X64-AN Quad, applications ensure trigger-to-image reliability by monitoring events and controlling transfer methods as described below:

Trigger Signal Validity

External trigger signal noise or glitches are easily ignored by the ACU with its programmable debounce control. A parameter is programmed for the minimum pulse duration considered as a valid external trigger pulse. Refer to "External Trigger and Strobe" on page 41 for more information.

Acquisition Events

Acquisition events are related to the acquisition module. They provide feedback on the image digitization phase. The following block diagram illustrates the acquisition process.

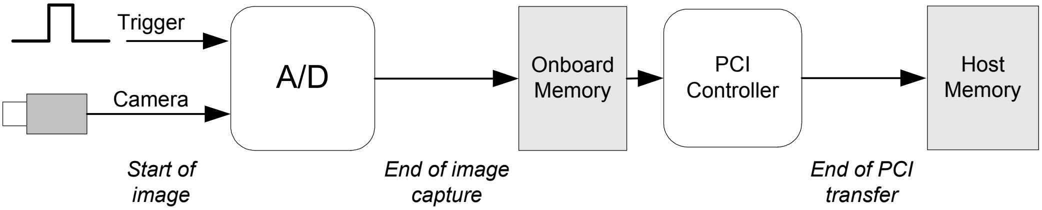


Figure 30: Acquisition/Transfer Interrupts

Event-Related Definitions

Interrupt	An interrupt is a signal sent by the X64-AN Quad board to the computer CPU which indicates an event on the frame grabber. The X64-AN Quad driver has excellent event reaction time since interrupts are processed inside an interrupt service routine (ISR) at kernel level.
Event	An event is a WIN32 object that can take two states: signaled and non-signaled. It is used for thread synchronization. In this context, an event is associated with an interrupt so that a WIN32 thread can be unblocked when the event it is waiting for gets signaled. For example, when an interrupt is received, the corresponding event is signaled and the thread waiting for this event resumes execution.

External Trigger (Used/Ignored)

The External Trigger event is generated when the external trigger pin is asserted, usually indicating the start of the acquisition process. There are 2 types of external trigger events: 'Used' or 'Ignored'. Following an external trigger, if the event generates a captured image, an External Trigger Used event will be generated (CORACQ_VAL_EVENT_TYPE_EXTERNAL_TRIGGER).

If there is no captured image, an External Trigger Ignored event will be generated (CORACQ_VAL_EVENT_TYPE_EXTERNAL_TRIGGER_IGNORED). An external trigger event will be ignored if the rate at which the events are received are higher than the possible frame rate of the camera.

On X64-AN Quad, the external trigger is protected by an opto-coupler. A minimum pulse width of 10µs is necessary to detect an active high trigger pulse while a minimum pulse width of 50µs is required for an active low trigger pulse. The X64-AN Quad is also equipped with signal debounce input circuit that allows the user to define the minimum acceptable pulse width via Sapera (CORACQ_PRM_EXTERNAL_TRIGGER_DURATION).

The region where an External Trigger will be ignored can be programmed using the 2 parameters CORACQ_PRM_EXT_TRIGGER_IGNORE_DELAY and CORACQ_PRM_CAM_CONTROL_DURING_READOUT.

Vertical Sync from Camera

The Vertical Sync event indicates a vertical sync has been detected. Note that this does not necessarily mean the image will be captured. For instance, if you have a free-running camera at 30 fps with external trigger enabled, you will get 30 events per second even though the X64-AN Quad waits for an external trigger to actually capture the next image. This allows the application program to independently count frames coming from the camera. The Sapera event value is CORACQ_VAL_EVENT_TYPE_VERTICAL_SYNC.

Horizontal Sync (Lock/Unlock)

The Horizontal Sync Lock/Unlock event indicates the state of the Analog to Digital Converter PLL with respect to the incoming horizontal sync. In order to digitize accurately, the PLL must be synchronized (i.e. be locked) to the incoming video HS. This interrupt is available when video is connected to the X64-AN Quad, whether it is acquiring images or not. In addition, a corresponding status flag can be read from the acquisition module. Typically the application first verifies the HS lock condition before starting an acquisition sequence.

The Sapera event values are CORACQ_VAL_EVENT_TYPE_HSYNC_LOCK and CORACQ_VAL_EVENT_TYPE_HSYNC_UNLOCK.

The Sapera status values are CORACQ_VAL_SIGNAL_HSYNC_LOCK and CORACQ_VAL_SIGNAL_HSYNC_UNLOCK.

Data Overflow

The Data Overflow event indicates that there is not enough bandwidth for the acquired data to flow. This is usually caused by limitations of the acquisition module. Since the X64-AN Quad can easily sustain onboard data transfers over 320MB, data overflow should never occur. The Sapera event value is CORACQ_VAL_EVENT_TYPE_DATA_OVERFLOW.

Frame Lost

The Frame Lost event indicates that an acquired image could not be transferred to onboard memory. An example of this case would be if there are no free onboard buffers available for the new image. This will usually be the case if the image transfer from onboard buffers to host PC memory cannot be sustained due to the PCI bus bandwidth. If multiple PCI bus master devices are active simultaneously, it is possible that the X64-AN Quad PCI controller can not transfer onboard buffers in time for the next acquired frame. In such a situation, an X64-AN Quad board with more memory would store more onboard frames without loss. The Sapera event value is CORACQ_VAL_EVENT_TYPE_FRAME_LOST.

Start/End of Field/Frame/Odd/Even

Acquisition events are available to indicate the start and end of field (odd or even), even field, or odd field (interlaced acquisition), and for the start and end of frame (progressive acquisition).The corresponding Sapera event values are:

CORACQ_VAL_EVENT_START_OF_FIELD, CORACQ_VAL_EVENT_START_OF_ODD,
CORACQ_VAL_EVENT_START_OF_EVEN, CORACQ_VAL_EVENT_START_OF_FRAME,
CORACQ_VAL_EVENT_END_OF_FIELD, CORACQ_VAL_EVENT_END_OF_ODD,
CORACQ_VAL_EVENT_END_OF_EVEN, CORACQ_VAL_EVENT_END_OF_FRAME,

Vertical Timeout

A vertical time event is generated if a vertical sync is not detected following an external/internal/software trigger within the period specified by CORACQ_PRM_VERTICAL_TIMEOUT_DELAY.

For analog cameras, if the WEN signal is used, the beginning of the WEN must be detected before the programmed delay expires. If syncing to blanking signals ,the end of the blanking signal must be detected before the programming delay expires.

The Sapera event value is CORACQ_VAL_EVENT_TYPE_VERTICAL_TIMEOUT.

Transfer Events

Transfer events are the ones related to the transfer module. Transfer events provide feedback on image transfer from onboard memory frame buffers to PC memory frame buffers.

Start of Frame

The Start of Frame event represents the beginning of a full frame transfer from onboard memory into PC memory. For interlaced video, there is one Start of Frame interrupt for each pair of fields. The Sapera event value is CORACQ_VAL_EVENT_TYPE_START_OF_FRAME.

Start of Field

The Start of Field event is only available for interlaced scan cameras. There is a Start of Field event at the beginning of each field transferred from onboard memory into PC memory (two per frame). The Sapera event value is CORACQ_VAL_EVENT_TYPE_START_OF_FIELD.

Start of Odd Field

The Start of Odd Field event is only available for interlaced scan cameras. There is a Start of Odd Field event at the beginning of each incoming odd field transferred from onboard memory into PC memory. The Sapera event value is CORACQ_VAL_EVENT_TYPE_START_OF_ODD.

Start of Even Field

The Start of Even Field event is only available for interlaced scan cameras. There is a Start of Even Field event at the beginning of each incoming even field transferred from onboard memory into PC memory. The Sapera event value is CORACQ_VAL_EVENT_TYPE_START_OF_EVEN.

End of Frame

The End of Frame event is generated when the last image pixel is transferred from onboard memory into PC memory. The Sapera event value is CORACQ_VAL_EVENT_TYPE_END_OF_FRAME.

End of Field

The End of Field event is only available for interlaced scan cameras. There is an End of Field event when the last field has been transferred from onboard memory into PC memory. The Sapera event value is CORACQ_VAL_EVENT_TYPE_END_OF_FIELD.

End of Odd Field

The End of Odd Field event is only available for interlaced scan cameras. There is an End of Odd Field event when the odd field has been transferred from onboard into PC memory. The Sapera event value is CORACQ_VAL_EVENT_TYPE_END_OF_ODD.

End of Even Field

The End of Even Field event is only available for interlaced scan cameras. There is an End of Even Field event when the even field has been transferred from onboard memory into PCI memory. The Sapera event value is CORACQ_VAL_EVENT_TYPE_END_OF_EVEN.

End of Transfer

The End of Transfer event is generated at the completion of the last image being transferred from onboard memory into PC memory. To complete a transfer, a stop must be issued to the transfer module (if transfers are already in progress). If a transfer of a fixed number of frames was requested, the transfer module will stop transfers automatically. The Sapera event value is CORACQ_VAL_EVENT_TYPE_END_OF_TRANSFER.

End of Line

The End of Line event is generated at the end of each line transferred from onboard memory into PC memory. Note that this event should only be used with very slow line rates. Standard RS170 cameras, with a line rate of 15kHz would generate too many events for the PC to follow. The Sapera event value is CORACQ_VAL_EVENT_TYPE_END_OF_LINE.

End of ‘n’ Lines

The End of ‘n’ Lines event is generated at the end of a group of ‘n’ lines transferred from onboard memory into PC memory. Note that this event should only be used with slow line rates. The Sapera event value is CORACQ_VAL_EVENT_TYPE_END_OF_NLINES.

Supported Transfer Cycling Methods

The X64-AN Quad supports the following transfer cycle modes which are either synchronous or asynchronous. These definitions are from the Sapera Basic Reference manual.

- **CORXFER_VAL_CYCLE_MODE_SYNCHRONOUS_WITH_TRASH**

Before cycling to the next buffer in the list, the transfer device will check the next buffer's state. If its state is full, the transfer will be done in the trash buffer which is defined as the last buffer in the list; otherwise, it will occur in the next buffer. After a transfer to the trash buffer is done, the transfer device will check again the state of the next buffer. If it is empty, it will transfer to this buffer otherwise it will transfer again to the trash buffer.

- **CORXFER_VAL_CYCLE_MODE_SYNCHRONOUS_NEXT_EMPTY_WITH_TRASH**

Before cycling to the next buffer in the list, the transfer device will check the next buffer's state. If its state is full, the next buffer will be skipped, and the transfer will be done in the trash buffer, which is defined as the last buffer in the list; otherwise it will occur in the next buffer. After a transfer to the trash is done, the transfer device will check the next buffer in the list, if its state is empty, it will transfer to this buffer otherwise it will skip it, and transfer again to the trash buffer.

- **CORXFER_VAL_CYCLE_MODE_ASYNCHRONOUS**

The transfer device cycles through all buffers in the list without concern about the buffer state.

Technical Reference

X64-AN Quad Board Specifications

Function	Description
Acquisition	
	Standard RS-170, RS-330, CCIR, and non-standard progressive scan providing composite video (non-standard progressive scan can be driven with external timing: HSYNC, VSYNC, and Frame Reset)
	Four analog video inputs, AC coupled and terminated to 75Ω
	Input video levels of 400mV to 1.2V supported
	8-bit A/D; Input pixel rates from 8MHz to 50MHz
	Pixel jitter: less than 2*ns*
	Simultaneous capture from any four synchronized or asynchronous cameras
	DC Restoration – programmable clamp pulse
	Partial scan mode
	Brightness & Contrast controls (Programmable Gain/Offset)
	Low-pass filter – jumper selectable
	Programmable time-base generator and programmable resolution, interlaced or non-interlaced

	horizontal period <= 255µsec:
	if (Pixel clock < 13 MHz) 2046 horizontal by 16,777,215 vertical
	if (Pixel clock > 13 MHz) 4094 horizontal by 16,777,215 vertical

	Horizontal: up to 4094 pixels in multiple of 4 pixels in slave-mode, 4080 in master-mode. Vertical: up to 16777215 lines in slave-mode and 65535 in master-mode.

Synchronization and timing control

Composite sync or Separate sync
H Sync range: 15 kHz to 100 kHz
V Sync range: 15 Hz to 120 kHz

Trigger input, opto-isolated TTL or RS-422
(optocoupler spec: 3.3 to 5 V input requiring a minimum current of 8mA)

Programmable trigger de-bounce delay from 1 to 255 microseconds

Four External Trigger inputs. Any one of them can be used to trigger acquisition from any camera.

Programmable trigger and strobe

2 Master Mode sync generators permit the genlocking of up to 2 different types of cameras

One strobe output per input

Outputs:
H sync, V sync (TTL up to 8 mA drive)
Trigger, strobe, and exposure (TTL up to 8mA drive)

Four onboard RS-232 COM ports for camera control
(mapped as host system COM ports)

Host transfers and data format

Pixel format: MONO8

Simultaneous transfer of up to four camera images into host memory

DMA engine supports typical sustained transfers up to 328MB (PCI-64), 120MB/second (PCI-32)

DMA engine supports maximum burst transfers up to 528MB (PCI-64), 132MB/second (PCI-32)

DMA engine supports scatter/gather

DMA engine supports de-interlacing images

External Signal Pins

Frame reset, VS, HS, WEN

TTL 3.3V (5V tolerant)

Source current 8mA, Sink current 8mA

On-board Processing

Input lookup-tables (256 x 8-bit) – following A/D

Area of Interest transfers

Output format: four 8-bit pixels per DWORD

Camera Power Source	
	Camera power via Hirose connector, +12 V @ 500 mA, fused protected
Board Power Requirements	
	+3.3 Volts: 2.62A (standby) – 2.74A (during acquisition) typical +5 Volts: 330mA typical +12 Volts: 150mA typical -12 Volts: 120mA typical -5 Volts: not used

Camera Compatibility

Go to the Teledyne DALSA Camera Database web page for the latest camera information at [http://www.teledynedalsa.com/mv/support/support.aspx].

Host System Requirements

The X64-AN Quad requires at minimum an Intel Pentium III or compatible computer system with a free PCI-32 or PCI-64 local bus slot supporting the PCI 3.3 volt specification.

Operating System Support

Either 32-bit or 64-bit versions of Windows XP, Windows Vista and Windows 7

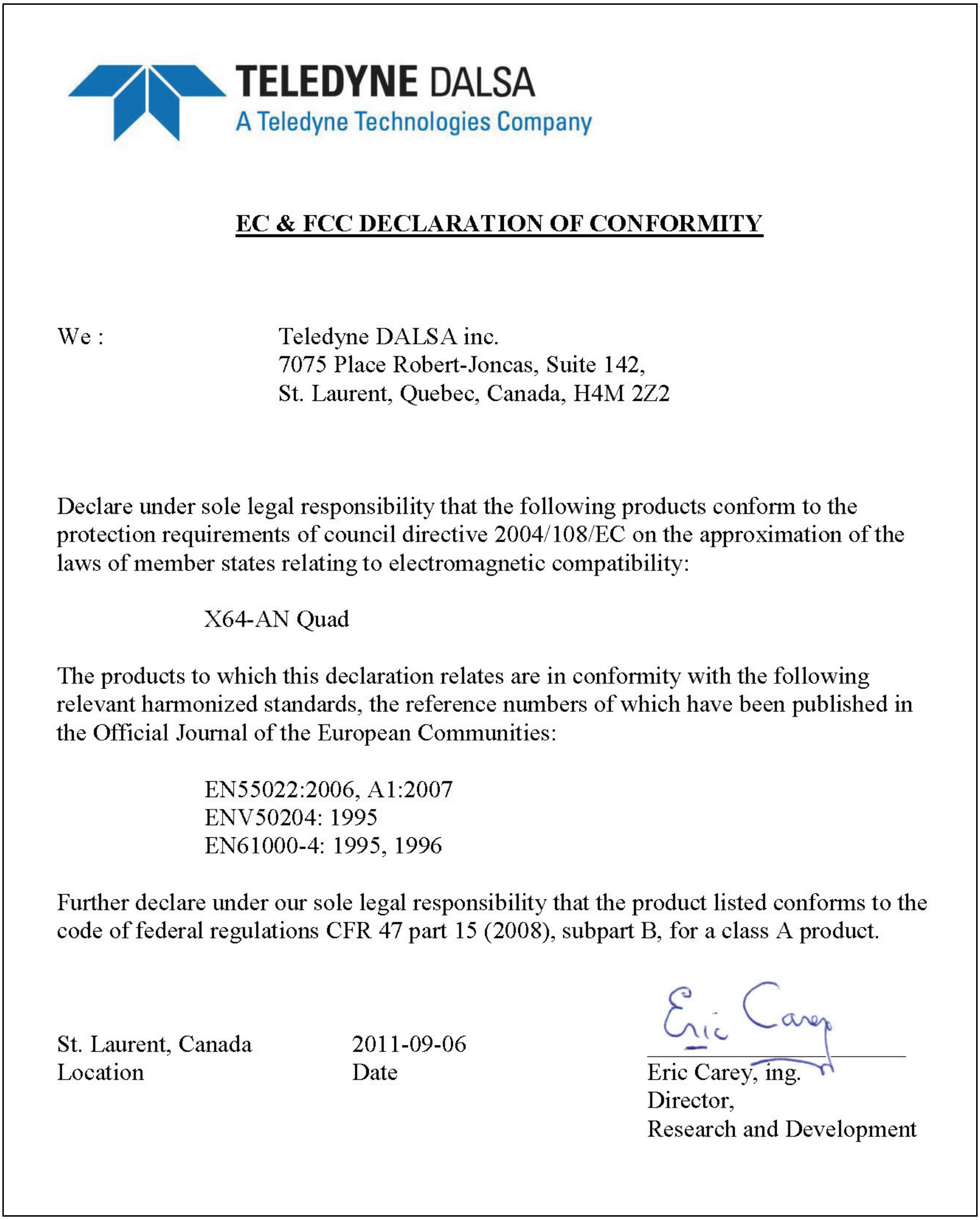
X64-AN Quad Physical Dimensions

Conforms to PCI full length PCB; approximately 12.25" W×4.125" H (31 cm W×10.5 cm H)

Environment

Ambient Temperature:	0° to 55° C (operation) -40° to 125° C (storage)
Relative Humidity:	5% to 95% non-condensing (operating) 0% to 95% (storage)

EMI Certifications



TELEDYNE DALSA
A Teledyne Technologies Company

EC & FCC DECLARATION OF CONFORMITY

We : Teledyne DALSA inc.
7075 Place Robert-Joncas, Suite 142,
St. Laurent, Quebec, Canada, H4M 2Z2

Declare under sole legal responsibility that the following products conform to the protection requirements of council directive 2004/108/EC on the approximation of the laws of member states relating to electromagnetic compatibility:

X64-AN Quad

The products to which this declaration relates are in conformity with the following relevant harmonized standards, the reference numbers of which have been published in the Official Journal of the European Communities:

EN55022:2006, A1:2007
ENV50204: 1995
EN61000-4: 1995, 1996

Further declare under our sole legal responsibility that the product listed conforms to the code of federal regulations CFR 47 part 15 (2008), subpart B, for a class A product.

		Eric Carey
St. Laurent, Canada	2011-09-06	
Location	Date	Eric Carey, ing. Director, Research and Development

X64-AN Quad Connector and Jumper Locations

Connector List

Descriptions for connectors and status LEDs follow the X64-AN Quad component layout drawings.

Connector	Description	Connector	Description
J5	DB9 – External Signal & Trigger inputs (see "J5 – Trigger Signals Connector" on page 65)	J17	PC power to camera interface (see "J17: Power Connector" on page 69)
J6, J7	Connectors for Hirose input module assembly	J15	Normal Operation (jumper on) Safe Start Mode (jumper off) (see "J15: Boot Recovery Mode" on page 70)
J8, J9, J10, J11	see "J8, J9, J10, J11: Input Low Pass Filter Select" on page 65	J21	Memory module socket
J19	Strobe outputs and Serial Ports (see "J19: Strobe & Com Ports" on page 69)	J16	X-I/O Interface using cable OC-IO0C-ANLVDS
D13, D14, D15, D16	Input Status LEDS (see "Acquisition Status LED" on page 67)	J13, J18, J22	Reserved

X64-AN Quad Board Component View

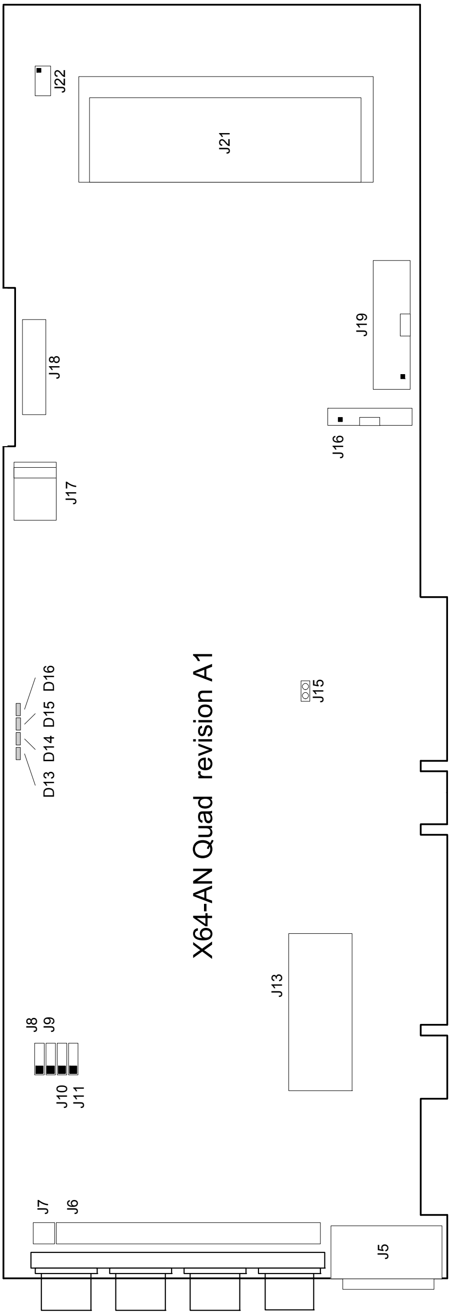


Figure 31: X64-AN Quad Board Component View

Connector Bracket End View

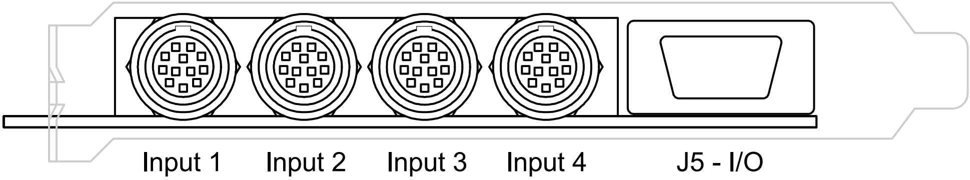


Figure 32: Connector Bracket

Hirose Input Connectors

The following drawing shows the female Hirose connector mounted on the X64-AN Quad as seen when looking at the board mounted in a computer. Pin descriptions follow.

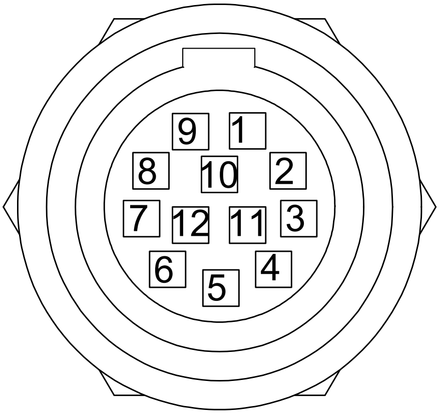


Figure 33: Hirose Pin Numbering

Hirose Pin Description

The X64-AN Quad features industrially secure Hirose connectors permitting a single cable connection to popular analog cameras (such as from Sony, Hitachi, and Pulnix). Signal pins are software configurable to easily support cameras as they become available. HD, VD, video, trigger, exposure, and their corresponding ground connections are configured for the pin-out required by the camera. Also available on the connector is a 12 volt fused power source for the camera (requires cable assembly cable OC-COMC-POW03 connected to J17 and an unused computer disk drive power connector).

Pin Number	X64-AN Quad	JAI	Sony, Hitachi	Pulnix
1	GND	GND	GND	GND
2	12v	12v	12v	12v
3	GND (video)	GND (video)	GND (video)	GND (video)
4	Video input	Video input	Video input	Video input
5	GND	GND	GND (HD I/O)	GND (Trigger)
6	camera control I/O *Programmable*	HD in	HD I/O	Trigger
7	camera control I/O *Programmable*	VD in	VD I/O	VD I/O
8	GND	GND	GND (Trigger output)	GND (HD I/O)
9	camera control I/O *Programmable*	NC/PCLK	Trigger output	HD I/O
10	camera control I/O *Programmable*	NC/WEN out	GND	GND
11	camera control I/O *Programmable*	NC/Ext Trig in	12v	
12	GND	GND	GND (VD I/O)	GND (VD I/O)

Note: X64-AN Quad programmable camera control I/O pins are typically defined by Sapera camera files. Camera files, as distributed by Teledyne DALSA or defined new by X64-AN Quad users, are configured with the Sapera CamExpert tool.

J8, J9, J10, J11: Input Low Pass Filter Select

The Input Low Pass Filter Select jumpers select the option of applying an input low pass filter to each input.

The following figure shows the jumper Enabled and Disabled (bypass) positions.

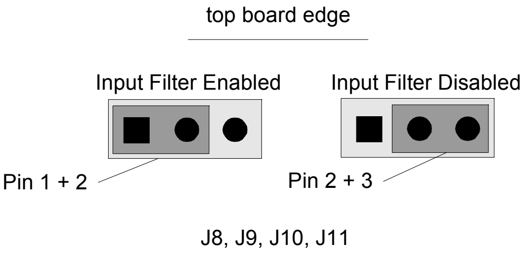


Figure 34: Input Low Pass Filter Selection Jumper

The default jumper position is set to Disabled for each of the four inputs (that is, pins 2 and 3 shorted).

Note: J8 (Input 1), J9 (Input 2), J10 (Input 3), J11 (Input 4)

J5 – Trigger Signals Connector

The following figure is the DB9 male connector view when looking at the X64-AN Quad connector bracket. The four X64-AN Quad trigger inputs use opto-coupler isolation circuits. A 650 ohm resistor is in series with the anode. Trigger input cable (number OC-VIPC-QDTRIG, "DB9 to four BNC") is available to simplify connecting to trigger signal sources.

See section “External Trigger” on page 41 for details on using the trigger inputs.

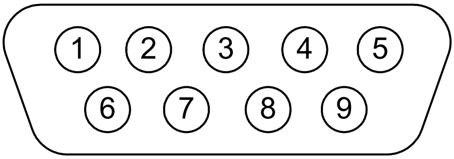


Figure 35: Trigger Inputs – DB9 Male Connector

The following figure is a simplified drawing of one opto-coupler trigger input. Trigger signals requirements are defined below.

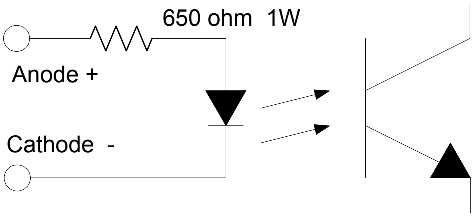


Figure 36: Trigger Input Opto-coupler

Pin Number	Description
1	Input 1 Trigger Input (anode: Ext_Trig+)
2	Input 2 Trigger Input (anode: Ext_Trig+)
3	Input 3 Trigger Input (anode: Ext_Trig+)
4	Input 4 Trigger Input (anode: Ext_Trig+)
5	Ground
6	Input 1's Trigger Input (cathode: Ext_Trig-)
7	Input 2's Trigger Input (cathode: Ext_Trig-)
8	Input 3's Trigger Input (cathode: Ext_Trig-)
9	Input 4's Trigger Input (cathode: Ext_Trig-)
Shell	Ground

Electrical parameters	Description	Value
$V_{IL\ max}$	Maximum voltage difference to turn OFF	0.8 V
$V_{IH\ min}$	Minimum voltage difference to turn ON	2 V
$I_{I\ min}$	Minimum input current to turn ON	2 mA
$I_{I\ max}$	Maximum input current to turn ON	50 mA
$t_{I\ min}$	Minimum input pulse width to turn ON	10 us
$V_{f\ max}$	Maximum forward voltage	24 V
$V_{r\ max}$	Maximum reverse voltage	-25 V

Note: TTL signals are approximately 0 and 5V, corresponding to logical 0 and 1, respectively. A standard TTL output can sink 16mA and could be used as a sink to drive an opto-coupled input. That is, +5V is connected to Ext_Trig+ and the sink trigger source is connected to Ext_Trig-.

Acquisition Status LED

The four status LEDs (one per input) provide visual feedback on the acquisition status. The LEDs are located on the top edge of the X64-AN Quad therefore require an open host system case to be visible (typical when doing system setup and verification).

D13 (input 1), D14 (input 2), D15 (input 3), D16 (input 4) have the following visual modes.

LED Status	Status Description
LED off	X64-AN Quad firmware is not loaded
Solid RED	Can not synchronize to input, (i.e. no video or unstable HS)
Solid Green	HS present and stable
Slow Flashing Green	HS and VS present and stable
Fast Flashing Green	Acquisition in progress

Serial Ports/Strobe Outputs Connector Bracket Assembly

The X64-AN Quad is equipped with a connector bracket assembly to allow easy interfacing to camera serial ports and strobe lights. Connect the 26-pin header connector to J19 on the X64-AN Quad. Mount the bracket to the computer case in a free slot position near the X64-AN Quad.

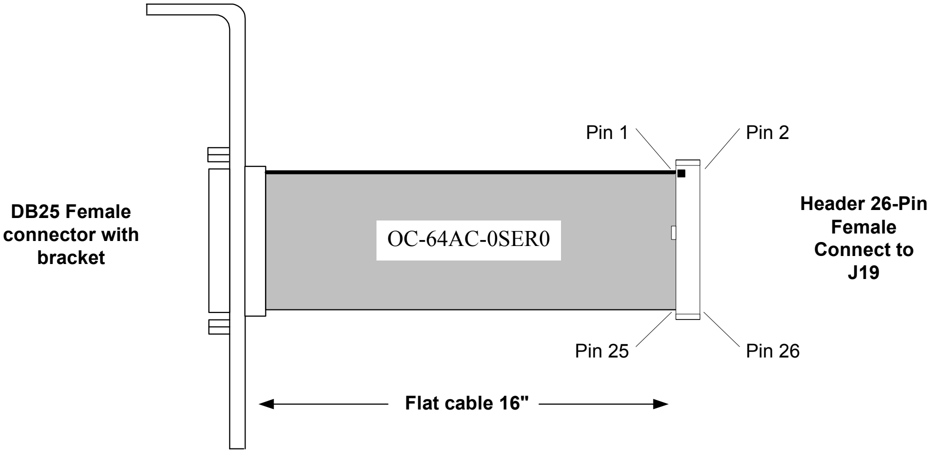

DB25 Female connector with bracket
OC-64AC-0SER0
Pin 1
Pin 2
Header 26-Pin Female Connect to J19
Pin 25
Pin 26
Flat cable 16"

Figure 37: OC-64AC-0SER0 serial port/strobe cable assembly

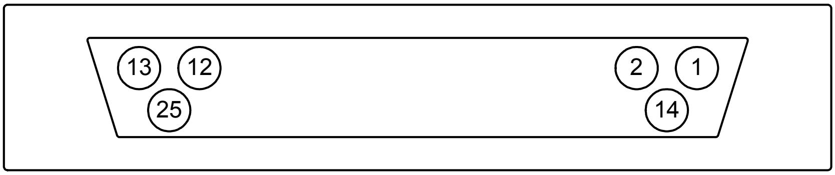


Figure 38: DB25 Female Connector End View

DB25 Pinout Description

Description	Odd Pins	Even Pins	Description
Ground	1	14	CAM1 serial RX
CAM1 serial TX	2	15	Ground
Strobe CAM1	3	16	reserved output
Ground	4	17	CAM2 serial RX
CAM2 serial TX	5	18	Ground
Strobe CAM2	6	19	reserved output
Ground	7	20	CAM3 serial RX
CAM3 serial TX	8	21	Ground
Strobe CAM3	9	22	reserved output
Ground	10	23	CAM4 serial RX
CAM4 serial TX	11	24	Ground
Strobe CAM4	12	25	reserved output
Ground	13		

J19: Strobe & Com Ports

The X64-AN Quad has one strobe output signal available per input. The polarity and pulse duration are programmable (up to 65 seconds). The following table defines the strobe pinout on J19. See section “Strobe” on page 44 for more information.

The X64-AN Quad has four serial ports for camera control. Each port can be mapped to a PC com port if required. See section “Serial Port” on page 45 for more information. The following table defines the serial port pinout on J19. Each port is made up of one transmit (TX)/receive (RX) pair plus ground.

2	4	...	24	26
1	3	...	23	25

Figure 39: J19 Pin Numbering

Description	Odd Pins	Even Pins	Description
Ground	1	2	CAM1 RS232_RX
CAM1 RS232_TX	3	4	Ground
Strobe CAM1	5	6	reserved output
Ground	7	8	CAM2 RS232_RX
CAM2 RS232_TX	9	10	Ground
Strobe CAM2	11	12	reserved output
Ground	13	14	CAM3 RS232_RX
CAM3 RS232_TX	15	16	Ground
Strobe CAM3	17	18	reserved output
Ground	19	20	CAM4 RS232_RX
CAM4 RS232_TX	21	22	Ground
Strobe CAM4	23	24	reserved output
Ground	25	26	NC

J17: Power Connector

A computer floppy disk power cable connects to J17 to provide 12V to cameras via the Hirose-12 input connectors. Camera power is fused (power-off reset). Use cable OC-COMC-POW03 to connect to an unused computer hard disk power supply cable.

J21: Memory Socket

SODIMM socket with 128 MB installed. Memory type or size changes are not supported. See "Memory Requirements with Area Scan Acquisitions" on page 30 for information on a possible error during frame buffer allocation.

J15: Boot Recovery Mode

- Default Mode: Shunt jumper is installed. The X64-AN Quad boots normally on system power up and is ready to execute image capture applications.
- Boot Recovery Mode: Shunt jumper is removed if any problems occurred while updating the X64 firmware. With the jumper off, reboot the computer and update the firmware again. When the update is complete, install the jumper and reboot the computer once again. (See "Recovering from a Firmware Update Error" on page 28).

J16: X-I/O Interface Connector

Use cable OC-IO0C-ANLVDS. See "Appendix: X-I/O Module Option" on page 99.

J13, J18, J22: Reserved

These connectors are for internal use or future product development. Signal descriptions are not publicly documented. No connections should be made due to the high risk of permanent damage to the X64-AN Quad.

Sapera LT

Sapera Server and Resources

The following table lists the Sapera Server available for X64-AN Quad. Note that a single server supports up to four cameras.

Servers		Resources			
Name	**Descriptio n**	**Type**	**Name**	**Inde x**	**Description**
X64-AN_1	X64-AN Quad	Acquisition	Analog Monochrome #1	0	Monochrome video Input #1
			Analog Monochrome #2	1	Monochrome video Input #2
			Analog Monochrome #3	2	Monochrome video Input #3
			Analog Monochrome #4	3	Monochrome video Input #4

Supported Sapera Parameters

The following table describes the Sapera parameters and values supported by X64-AN Quad. Refer to *Sapera Acquisition Parameters Reference* manual for a thorough description of each parameter.

Camera Related Capabilities	Values
CORACQ_CAP_CONNECTOR_TYPE	CORACQ_VAL_CONNECTOR_TYPE_HIROSE12 (0x1)
CORACQ_CAP_CONNECTOR_CAMLINK	
Pin – 01, Pin – 05, Pin – 08, Pin - 12	CORACQ_VAL_SIGNAL_NAME_GND (0x4000)
Pin – 02	CORACQ_VAL_SIGNAL_NAME_POWER_12V (0x8000)
Pin – 03	CORACQ_VAL_SIGNAL_NAME_VIDEO_GND (0x20000)
Pin – 04	CORACQ_VAL_SIGNAL_NAME_VIDEO (0x10000)
Pin – 06	CORACQ_VAL_SIGNAL_NAME_NO_CONNECT (0x1) CORACQ_VAL_SIGNAL_NAME_HD (0x2) CORACQ_VAL_SIGNAL_NAME_PULSE0 (0x8) CORACQ_VAL_SIGNAL_NAME_PULSE1 (0x10) CORACQ_VAL_SIGNAL_NAME_WEN (0x100)
Pin – 07	CORACQ_VAL_SIGNAL_NAME_NO_CONNECT (0x1) CORACQ_VAL_SIGNAL_NAME_VD (0x4) CORACQ_VAL_SIGNAL_NAME_PULSE0 (0x8) CORACQ_VAL_SIGNAL_NAME_PULSE1 (0x10) CORACQ_VAL_SIGNAL_NAME_WEN (0x100)
Pin – 09	CORACQ_VAL_SIGNAL_NAME_NO_CONNECT (0x1) CORACQ_VAL_SIGNAL_NAME_HD (0x2) CORACQ_VAL_SIGNAL_NAME_VD (0x4) CORACQ_VAL_SIGNAL_NAME_PULSE0 (0x8) CORACQ_VAL_SIGNAL_NAME_PULSE1 (0x10) CORACQ_VAL_SIGNAL_NAME_WEN (0x100)
Pin – 10	CORACQ_VAL_SIGNAL_NAME_NO_CONNECT (0x1) CORACQ_VAL_SIGNAL_NAME_PULSE0 (0x8) CORACQ_VAL_SIGNAL_NAME_WEN (0x100)
Pin - 11	CORACQ_VAL_SIGNAL_NAME_NO_CONNECT (0x1) CORACQ_VAL_SIGNAL_NAME_PULSE0 (0x8) CORACQ_VAL_SIGNAL_NAME_PULSE1 (0x10) CORACQ_VAL_SIGNAL_NAME_WEN (0x100)

CAMERA PARAMETERS	Values
CORACQ_PRM_CAM_CONTROL_DURING_READOUT	TRUE FALSE
CORACQ_PRM_CAM_RESET_DURATION	Min: 1 µs Max: 65535000 µs Step: 1 µs
CORACQ_PRM_CAM_RESET_METHOD	CORACQ_VAL_CAM_RESET_METHOD_1

CORACQ_PRM_CAM_RESET_POLARITY	CORACQ_VAL_ACTIVE_LOW CORACQ_VAL_ACTIVE_HIGH
CORACQ_PRM_CAM_TIME_INTEGRATE_DURATION_MAX	4294967295 µs
CORACQ_PRM_CAM_TIME_INTEGRATE_DURATION_MIN	0 µs
CORACQ_PRM_CAM_TRIGGER_DURATION	Min: 1 µs Max: 65535000 µs Step: 1 µs
CORACQ_PRM_CAM_TRIGGER_METHOD	CORACQ_VAL_CAM_TRIGGER_METHOD_1 CORACQ_VAL_CAM_TRIGGER_METHOD_2
CORACQ_PRM_CAM_TRIGGER_POLARITY	CORACQ_VAL_ACTIVE_LOW CORACQ_VAL_ACTIVE_HIGH
CORACQ_PRM_CHANNEL	CORACQ_VAL_CHANNEL_SINGLE
CORACQ_PRM_CHANNELS_ORDER	CORACQ_VAL_CHANNELS_ORDER_NORMAL
CORACQ_PRM_COUPLING	CORACQ_VAL_COUPLING_AC
CORACQ_PRM_FIELD_ORDER	CORACQ_VAL_FIELD_ORDER_NEXT_FIELD CORACQ_VAL_FIELD_ORDER_EVEN_ODD CORACQ_VAL_FIELD_ORDER_ODD_EVEN
CORACQ_PRM_FRAME	CORACQ_VAL_FRAME_INTERLACED CORACQ_VAL_FRAME_PROGRESSIVE
CORACQ_PRM_HACTIVE	Min: 1 pixel Max: 4095 pixel Step: 1 pixel
CORACQ_PRM_HBACK_PORCH	Min: 0 pixel Max: 4095 pixel Step: 1 pixel
CORACQ_PRM_HFRONT_PORCH	Min: 0 pixel Max: 4095 pixel Step: 1 pixel
CORACQ_PRM_HSYNC	Min: 1 pixel Max: 510 pixel Step: 1 pixel
CORACQ_PRM_HSYNC_POLARITY	CORACQ_VAL_ACTIVE_LOW CORACQ_VAL_ACTIVE_HIGH
CORACQ_PRM_INTERFACE	CORACQ_VAL_INTERFACE_ANALOG
CORACQ_PRM_PIXEL_CLK_11	Min: 1 Hz Max: (2 ** 32) – 1 Hz Step: 1 Hz
CORACQ_PRM_PIXEL_CLK_DETECTION	CORACQ_VAL_RISING_EDGE
CORACQ_PRM_PIXEL_CLK_INT	Min: 8000000 Hz Max: 50000000 Hz Step: 1 Hz
CORACQ_PRM_PIXEL_CLK_SRC	CORACQ_VAL_PIXEL_CLK_SRC_INT
CORACQ_PRM_PIXEL_DEPTH	8
CORACQ_PRM_SCAN	CORACQ_VAL_SCAN_AREA
CORACQ_PRM_SIGNAL	CORACQ_VAL_SIGNAL_SINGLE_ENDED
CORACQ_PRM_SYNC	CORACQ_VAL_SYNC_COMP_VIDEO CORACQ_VAL_SYNC_SEP_SYNC CORACQ_VAL_SYNC_INT_SYNC
CORACQ_PRM_TAP_1_DIRECTION	CORACQ_VAL_TAP_DIRECTION_FROM_TOP CORACQ_VAL_TAP_DIRECTION_UD CORACQ_VAL_TAP_DIRECTION_LR

CORACQ_PRM_TAP_OUTPUT	CORACQ_VAL_TAP_OUTPUT_SEGMENTED
CORACQ_PRM_TAPS	1
CORACQ_PRM_TIME_INTEGRATE_METHOD	CORACQ_VAL_TIME_INTEGRATE_METHOD_1 CORACQ_VAL_TIME_INTEGRATE_METHOD_2 CORACQ_VAL_TIME_INTEGRATE_METHOD_3 CORACQ_VAL_TIME_INTEGRATE_METHOD_4 CORACQ_VAL_TIME_INTEGRATE_METHOD_5 CORACQ_VAL_TIME_INTEGRATE_METHOD_6 CORACQ_VAL_TIME_INTEGRATE_METHOD_7
CORACQ_PRM_TIME_INTEGRATE_PULSE0_DELAY	Min: 0 µs Max: 65535000 µs Step: 1 µs
CORACQ_PRM_TIME_INTEGRATE_PULSE0_DURATION	Min: 1 µs Max: 65535000 µs Step: 1 µs
CORACQ_PRM_TIME_INTEGRATE_PULSE0_POLARITY	CORACQ_VAL_ACTIVE_LOW CORACQ_VAL_ACTIVE_HIGH
CORACQ_PRM_TIME_INTEGRATE_PULSE1_DELAY	Min: 0 µs Max: 65535000 µs Step: 1 µs
CORACQ_PRM_TIME_INTEGRATE_PULSE1_DURATION	Min: 1 µs Max: 65535000 µs Step: 1 µs
CORACQ_PRM_TIME_INTEGRATE_PULSE1_POLARITY	CORACQ_VAL_ACTIVE_LOW CORACQ_VAL_ACTIVE_HIGH
CORACQ_PRM_VACTIVE	Min: 1 line Max: 16777215 line Step: 1 line
CORACQ_PRM_VBACK_INVALID	Min: 0 line Max: 16777215 lineStep 1 line
CORACQ_PRM_VBACK_PORCH	Min: 0 line Max: 65535 line Step: 1 line
CORACQ_PRM_VFRONT_PORCH	Min: 0 line Max: 65535 lineStep: 1 line
CORACQ_PRM_VIDEO	CORACQ_VAL_VIDEO_MONO
CORACQ_PRM_VIDEO_LEVEL_MAX	>= CORACQ_PRM_VIDEO_LEVEL_MIN
CORACQ_PRM_VIDEO_LEVEL_MIN	<= CORACQ_PRM_VIDEO_LEVEL_MAX
CORACQ_PRM_VIDEO_STD	CORACQ_VAL_VIDEO_STD_NON_STD CORACQ_VAL_VIDEO_STD_RS170_NTSC CORACQ_VAL_VIDEO_STD_CCIR_PAL CORACQ_VAL_VIDEO_STD_SECAM
CORACQ_PRM_VSYNC	Min: 1 line Max: 255 line Step: 1 line
CORACQ_PRM_VSYNC_POLARITY	CORACQ_VAL_ACTIVE_LOW CORACQ_VAL_ACTIVE_HIGH
CORACQ_PRM_WEN_POLARITY	CORACQ_VAL_ACTIVE_LOW CORACQ_VAL_ACTIVE_HIGH

VIC PARAMETERS	Values
CORACQ_PRM_BIT_ORDERING	CORACQ_VAL_BIT_ORDERING_STD

CORACQ_PRM_BRIGHTNESS	Min: -20000 (-20%) Max: 29000 (29%) Step: 196 (0.196%)
CORACQ_PRM_CAM_CONTROL_PULSE0_HD_ALIGN	TRUE FALSE
CORACQ_PRM_CAM_CONTROL_PULSE1_HD_ALIGN	TRUE FALSE
CORACQ_PRM_CAM_RESET_DELAY	Min: 0 µs Max: 65535000 µs Step: 1 µs
CORACQ_PRM_CAM_RESET_ENABLE	TRUE FALSE
CORACQ_PRM_CAM_TRIGGER_DELAY	Min: 0 µs Max: 65535000 µs Step: 1 µs
CORACQ_PRM_CAM_TRIGGER_ENABLE	TRUE FALSE
CORACQ_PRM_CAMSEL	0
CORACQ_PRM_CONTRAST	Min: 60000 (60%) Max: 170000 (170%) Step: 1000 (1%)
CORACQ_PRM_CROP_HEIGHT	Min: 1 line Max: 16777215 lineStep: 1 line
CORACQ_PRM_CROP_LEFT	Min: 0 pixel Max: 16777215 pixel Step: 8 pixel
CORACQ_PRM_CROP_TOP	Min: 0 line Max: 16777215 line Step: 1 line
CORACQ_PRM_CROP_WIDTH	Min: 16 pixel Max: 16777215 pixel Step: 16 pixel
CORACQ_PRM_DC_REST_MODE	CORACQ_VAL_DC_REST_MODE_AUTO CORACQ_VAL_DC_REST_MODE_ON CORACQ_VAL_DC_REST_MODE_OFF
CORACQ_PRM_DC_REST_START	Min: 0 pixel Max: 2047 pixel Step: 1 pixel
CORACQ_PRM_DC_REST_WIDTH	Min: 0 pixel Max: 2047 pixel Step: 1 pixel
CORACQ_PRM_DECIMATE_METHOD	CORACQ_VAL_DECIMATE_DISABLE CORACQ_VAL_DECIMATE_ODD CORACQ_VAL_DECIMATE_EVEN
CORACQ_PRM_EXT_TRIGGER_DETECTION	CORACQ_VAL_RISING_EDGE CORACQ_VAL_FALLING_EDGE CORACQ_VAL_ACTIVE_LOW CORACQ_VAL_ACTIVE_HIGH
CORACQ_PRM_EXT_TRIGGER_DURATION	Min: 0 µs Max: 65535 µs Step: 1 µs
CORACQ_PRM_EXT_TRIGGER_ENABLE	CORACQ_VAL_EXT_TRIGGER_OFF CORACQ_VAL_EXT_TRIGGER_ON

CORACQ_PRM_EXT_TRIGGER_FRAME_COUNT	Min: 1 frame Max: 65535 frame Step: 1 frame
CORACQ_PRM_EXT_TRIGGER_LEVEL	CORACQ_VAL_LEVEL_TTL
CORACQ_PRM_EXT_TRIGGER_SOURCE	0: automatic selection. Use same trigger number as the acquisition module index. 1: Trigger Input #1 2: Trigger Input #2 3: Trigger Input #3 4: Trigger Input #4
CORACQ_PRM_FLIP	CORACQ_VAL_FLIP_HORZ
CORACQ_PRM_INT_FRAME_TRIGGER_ENABLE_	TRUE FALSE
CORACQ_PRM_INT_FRAME_TRIGGER_FREQ	Min: 1 Hz Max: 1073741823 Hz Step: 1 Hz
CORACQ_PRM_LUT_ENABLE	TRUE FALSE
CORACQ_PRM_LUT_FORMAT	CORACQ_VAL_OUTPUT__FORMAT_MONO8
CORACQ_PRM_LUT_MAX	1
CORACQ_PRM_LUT_NENTRIES	256
CORACQ_PRM_LUT_NUMBER	0
CORACQ_PRM_MASTER_MODE	CORACQ_VAL_MASTER_MODE_DISABLE CORACQ_VAL_MASTER_MODE_HSYNC_VSYNC CORACQ_VAL_MASTER_MODE_HSYNC
CORACQ_PRM_MASTER_MODE_HSYNC_POLARITY	CORACQ_VAL_ACTIVE_LOW CORACQ_VAL_ACTIVE_HIGH
CORACQ_PRM_MASTER_MODE_VSYNC_POLARITY	CORACQ_VAL_ACTIVE_LOW CORACQ_VAL_ACTIVE_HIGH
CORACQ_PRM_OUTPUT_FORMAT	CORACQ_VAL_OUTPUT_FORMAT_MONO8
CORACQ_PRM_SNAP_COUNT	Min: 1 frame Max: 65535 frame Step: 1 frame
CORACQ_PRM_STROBE_DELAY	Min: 0 µs Max: 65535000 µs Step: 1 µs
CORACQ_PRM_STROBE_DELAY_2	Min: 0 µs Max: 65535000 µs Step: 1 µs
CORACQ_PRM_STROBE_DURATION	Min: 1 µs Max: 65535000 µs Step: 1 µs
CORACQ_PRM_STROBE_ENABLE	TRUE FALSE
CORACQ_PRM_STROBE_METHOD	CORACQ_VAL_STROBE_METHOD_1 CORACQ_VAL_STROBE_METHOD_2 CORACQ_VAL_STROBE_METHOD_4
CORACQ_PRM_STROBE_LEVEL	CORACQ_VAL_LEVEL_TTL
CORACQ_PRM_STROBE_POLARITY	CORACQ_VAL_ACTIVE_LOW CORACQ_VAL_ACTIVE_HIGH

CORACQ_PRM_TIME_INTEGRATE_DELAY	Min: 0 µs Max: 65535000 µs Step: 1 µs
CORACQ_PRM_TIME_INTEGRATE_DURATION	Min: 0 µs Max: 65535000 µs Step: 1 µs
CORACQ_PRM_TIME_INTEGRATE_ENABLE	TRUE FALSE
CORACQ_PRM_VERTICAL_TIMEOUT_DELAY	Min: 0 µs Max: 16383000 µs Step: 1 µs
CORACQ_PRM_WEN_ENABLE	TRUE FALSE

ACQUISITION PARAMETERS	Values
CORACQ_PRM_EVENT_TYPE	CORACQ_VAL_EVENT_TYPE_VERTICAL_TIMEOUT CORACQ_VAL_EVENT_TYPE_HSYNC_LOCK CORACQ_VAL_EVENT_TYPE_HSYNC_UNLOCK CORACQ_VAL_EVENT_TYPE_EXTERNAL_TRIGGER_IGNORED CORACQ_VAL_EVENT_TYPE_DATA_OVERFLOW CORACQ_VAL_EVENT_TYPE_FRAME_LOST CORACQ_VAL_EVENT_TYPE_START_OF_FIELD CORACQ_VAL_EVENT_TYPE_START_OF_ODD CORACQ_VAL_EVENT_TYPE_START_OF_EVEN CORACQ_VAL_EVENT_TYPE_START_OF_FRAME CORACQ_VAL_EVENT_TYPE_END_OF_FIELD CORACQ_VAL_EVENT_TYPE_END_OF_ODD CORACQ_VAL_EVENT_TYPE_END_OF_EVEN CORACQ_VAL_EVENT_TYPE_END_OF_FRAME CORACQ_VAL_EVENT_TYPE_EXTERNAL_TRIGGER CORACQ_VAL_EVENT_TYPE_VERTICAL_SYNC
CORACQ_PRM_SIGNAL_STATUS	CORACQ_VAL_SIGNAL_HSYNC_PRESENT CORACQ_VAL_SIGNAL_VSYNC_PRESENT CORACQ_VAL_SIGNAL_HSYNC_LOCK CORACQ_VAL_SIGNAL_POWER_PRESENT

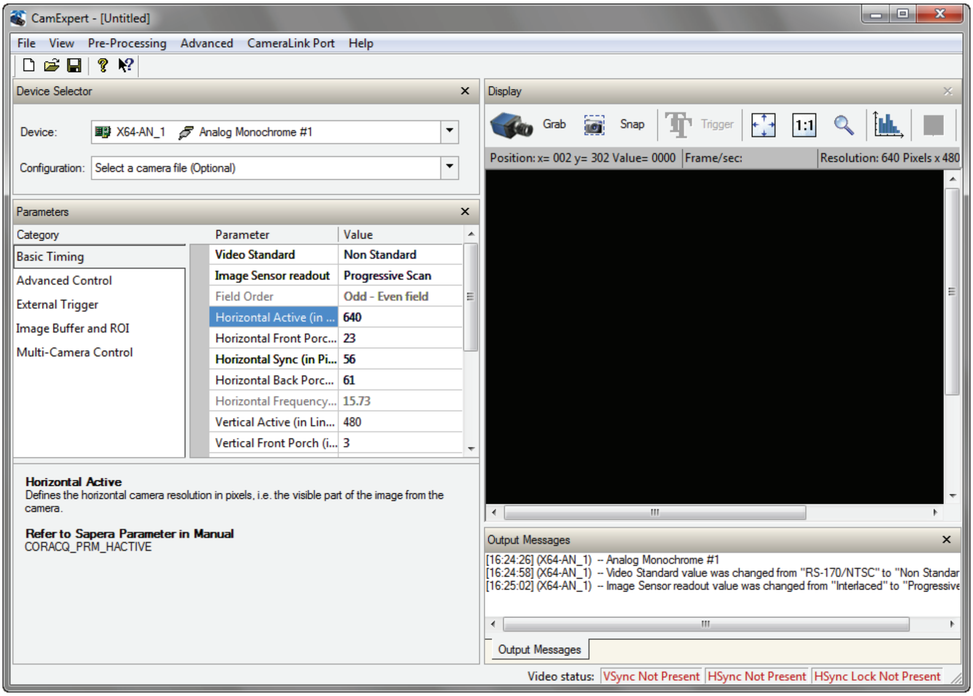
TRANSFER PARAMETERS	Values
CORXFER_PRM_EVENT_TYPE	CORXFER_VAL_EVENT_TYPE_START_OF_FIELD CORXFER_VAL_EVENT_TYPE_START_OF_ODD CORXFER_VAL_EVENT_TYPE_START_OF_EVEN CORXFER_VAL_EVENT_TYPE_START_OF_FRAME CORXFER_VAL_EVENT_TYPE_END_OF_FIELD CORXFER_VAL_EVENT_TYPE_END_OF_ODD CORXFER_VAL_EVENT_TYPE_END_OF_EVEN CORXFER_VAL_EVENT_TYPE_END_OF_FRAME CORXFER_VAL_EVENT_TYPE_END_OF_LINE CORXFER_VAL_EVENT_TYPE_END_OF_NLINES CORXFER_VAL_EVENT_TYPE_END_OF_TRANSFER

Using Sapera CamExpert with X64-AN Quad

CamExpert is the camera interfacing tool for frame grabber boards supported by the Sapera library. CamExpert generates the Sapera camera configuration file (*yourcamera.ccf*) based on timing and control parameters entered. For backward compatibility with previous versions of Sapera, CamExpert also reads and writes the *.cca and *.cvi camera parameter files.

An important component of CamExpert is its live acquisition display window which allows immediate verification of timing or control parameters without the need to run a separate acquisition program.

For context sensitive help, click on the ↖? button then click on a camera configuration parameter. A short description of the configuration parameter will be shown in a popup. Click on the ? button to open the help file for more descriptive information on CamExpert.



Device: X64-AN_1 Analog Monochrome #1

Configuration: Select a camera file (Optional)

Category: Basic Timing, Advanced Control, External Trigger, Image Buffer and ROI, Multi-Camera Control

Parameter	Value
Video Standard	Non Standard
Image Sensor readout	Progressive Scan
Field Order	Odd - Even field
Horizontal Active (in ...	640
Horizontal Front Porc...	23
Horizontal Sync (in Pi...	56
Horizontal Back Porc...	61
Horizontal Frequency...	15.73
Vertical Active (in Lin...	480
Vertical Front Porch (i...	3

Horizontal Active
Defines the horizontal camera resolution in pixels, i.e. the visible part of the image from the camera.

Refer to Sapera Parameter in Manual
CORACQ_PRM_HACTIVE

Position: x= 002 y= 302 Value= 0000 | Frame/sec: | Resolution: 640 Pixels x 480

```
[16:24:26] (X64-AN_1) -- Analog Monochrome #1
[16:24:58] (X64-AN_1) -- Video Standard value was changed from "RS-170/NTSC" to "Non Standar
[16:25:02] (X64-AN_1) -- Image Sensor readout value was changed from "Interlaced" to "Progressive
```

Video status: VSync Not Present | HSync Not Present | HSync Lock Not Present

Figure 40: CamExpert Tool

The central section of CamExpert provides access to the various X64-AN Quad Sapera parameters. It is divided into four or five tabs (dependent on the board capabilities).

Basic Timing Parameters	Basic parameters used to define the timing of the camera. This includes the vertical, horizontal, and pixel clock frequency. This tab is sufficient to configure a free-running camera.
Advanced Control Parameters	Advanced parameters used to configure camera control mode and strobe output. Also provides analog signal conditioning (brightness, contrast, DC restoration, etc.)
External Trigger Parameters	Parameters to configure the external trigger characteristics.
Image Buffer and ROI Parameters	Control of the host buffer dimension and format.
Multi-Camera Control Parameters	Provides camera selection in multi-camera modes when applicable.

Camera Interfacing Check List

Before undertaking the task of interfacing a camera from scratch with CamExpert:

- Confirm that Teledyne DALSA has not already published an application note with camera files [http://www.teledynedalsa.com/mv/support/support.aspx].
- Confirm that Sapera does not already have a .cca file for your camera installed on your hard disk. If there is a .cca file supplied with Sapera, then use CamExpert to automatically generate the .ccf file with default parameter values matching the frame grabber capabilities.
- Check if the Sapera installation has a similar type of camera file. A similar .cca file can be loaded into CamExpert where it is modified to match timing and operating parameters for your camera, and lastly save them as Camera Configuration file (.ccf).
- Finally, if your camera type has never been interfaced, run CamExpert after installing Sapera and the acquisition board driver, select the board acquisition server, and manually enter all camera parameters.

Camera Files Distributed with Sapera

The Sapera distribution CDROM includes camera files for a selection of supported analog cameras. Using the Sapera CamExpert program, you may use the camera files (CCA) provided to generate a camera configuration file (CCF) that describes the desired camera and frame grabber configuration.

Sapera Camera Application Library

Teledyne DALSA continually updates a camera application library composed of application information and prepared camera files. Refer to the camera search utility on the Teledyne DALSA web site [http://www.teledynedalsa.com/mv/support/support.aspx] for application notes. Camera files are ASCII text and can be read with Notepad.

Overview of Sapera Acquisition Parameter Files (*.ccf or *.cca/*.cvi)

Concepts and Differences between the Camera Parameter Files

There are two components to the legacy Sapera acquisition parameter file set: CCA files (also called cam files) and CVI files (also called VIC files, i.e., video input conditioning). The files store video-signal parameters (CCA) and video conditioning parameters (CVI), which in turn simplifies programming the frame grabber acquisition hardware for the camera in use. **Sapera LT 5.0** introduces a new camera configuration file (**CCF**) that combines the CCA and CVI files into one file.

Typically, a camera application will use a CCF file per camera operating mode (or one CCA file in conjunction with several CVI files, where each CVI file defines a specific camera operating mode). An application can also have multiple CCF files so as to support different image format modes supported by the camera or sensor (such as image binning or variable ROI).

CCF File Details

Files using the ".CCF" extension (Camera Configuration file) are essentially the camera (CCA) and frame grabber (CVI) parameters grouped into one file for easier configuration file management. This is the default Camera Configuration file used with Sapera LT 5.0 and the CamExpert utility.

CCA File Details

Teledyne DALSA distributes camera files using the ".CCA" extension that contain all parameters describing the camera video signal characteristics and operation modes (that is, what the camera outputs). The Sapera parameter groups located within the file are:

- Video format and pixel definitions.
- Video resolution (pixel rate, pixels per line, and lines per frame).
- Synchronization source and timings.
- Channels/Taps configuration.
- Supported camera modes and related parameters.
- External hardware signal assignment.

CVI File Details

Legacy files using the ".CVI" extension (Camera Video files) contain all operating parameters related to the frame grabber board, that is, what the frame grabber can actually do with camera controls or incoming video. The Sapera parameter groups located within the file:

- Activates and sets any supported camera control mode or control variable.
- Defines the integration mode and duration.
- Defines the strobe output control.
- Allocates the frame grabber transfer ROI, the host video buffer size and buffer type (RGB888, RGB101010, MONO8, MONO16).

- Configuration of line/frame trigger parameters such as source (internal via the frame grabber /external via some outside event), electrical format (TTL, LVDS, OPTO-isolated), and signal active edge or level characterization.

CamExpert Example: Interfacing the JAI CV-A11

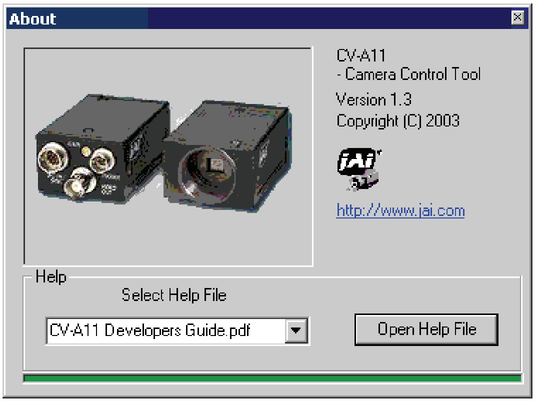
The CamExpert utility is presented by using the example of interfacing the JAI CV-A11 progressive scan camera to the X64-AN Quad. The major steps covered are:

- Camera interfacing check list
- CamExpert interfacing outline
- JAI CV-A11 in free run exposure mode
- JAI CV-A11 in Partial Scan mode
- JAI CV-A11 in Edge Pre-select (trigger) mode
- JAI CV-A11 in Time Integration mode

Download the user's manual directly from the JAI web site [http://www.jai.com/].

Interfacing Outline

- Sapera and the X64-AN Quad device driver are installed.
- Check for an existing CCA file, distributed with Sapera, that will provide camera timing parameters. For the JAI CV-A11 the supplied file is "J_CV-A11.cca ".
- Check the Teledyne DALSA web site for a published application note with camera files [http://www.teledynedalsa.com/mv/support/support.aspx].
- Connect the interface cables for video and serial port control to the camera. The X64-AN Quad provides power to the camera via the video cable.
- Run the JAI CV-A11 camera control tool to set the camera in free run mode. This will simplify testing the camera file timing parameters. The following screen image shows the control tool version used within this document.



CV-A11
- Camera Control Tool
Version 1.3
Copyright (C) 2003
http://www.jai.com

Select Help File: CV-A11 Developers Guide.pdf

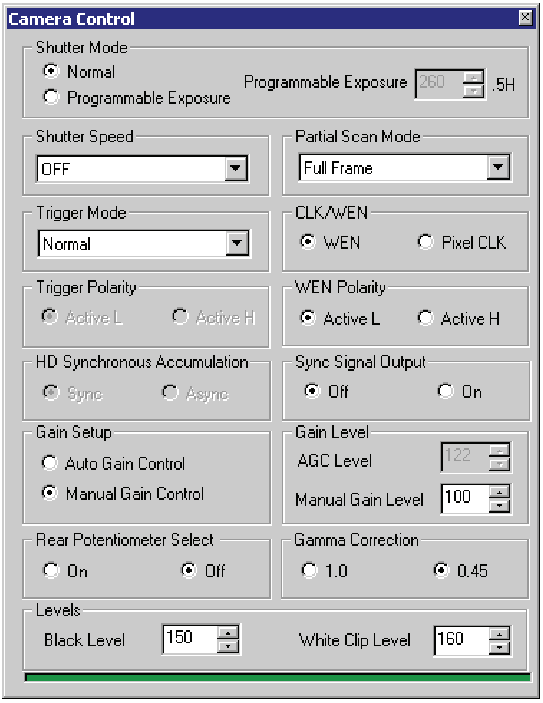
Figure 41: JAI CV-A11 Control Tool Info

- Run CamExpert, load or set camera timing parameters and capabilities, and then test a live grab. Save the camera file for the default free run mode.
- Use the JAI CV-A11 camera control tool to set the camera in alternative control modes such as time integration or reduced vertical size-higher frame rate modes.
- Configure CamExpert for the mode setup via the camera control tool. Test variations of these other modes. Save new camera files for these modes when satisfied.

step 1: JAI CV-A11 in Free Run Mode

This section illustrates interfacing the JAI CV-A11 in free run mode.

- Run the JAI camera control application. The factory defaults typically place the camera in free run mode as shown in the following screen image. Note that this example used a camera with an adjustable iris, thus the camera gain was set to manual. The JAI CV-A11 camera control tool can remain running.



Camera Control

Shutter Mode
● Normal
○ Programmable Exposure
Programmable Exposure 260 .5H

Shutter Speed
OFF

Partial Scan Mode
Full Frame

Trigger Mode
Normal

CLK/WEN
● WEN ○ Pixel CLK

Trigger Polarity
● Active L ○ Active H

WEN Polarity
● Active L ○ Active H

HD Synchronous Accumulation
● Sync ○ Async

Sync Signal Output
● Off ○ On

Gain Setup
○ Auto Gain Control
● Manual Gain Control

Gain Level
AGC Level 122
Manual Gain Level 100

Rear Potentiometer Select
○ On ● Off

Gamma Correction
○ 1.0 ● 0.45

Levels
Black Level 150
White Clip Level 160

Figure 42: JAI CV-A11 Camera Control Tool

Run Sapera CamExpert

- From the Windows start menu run the Sapera CamExpert program.

Start •All Programs • Teledyne DALSA • Sapera LT • CamExpert

- The CamExpert **Board** window shows the available Sapera acquisition devices where the X64-AN Quad monochrome input #1 is selected by default.
- Load the Sapera supplied camera file for the JAI CV-A11. CamExpert is now configured for JAI CV-A11 in free run mode. The following screen shot shows the CamExpert camera file selection menu with the JAI CV-A11 selected.

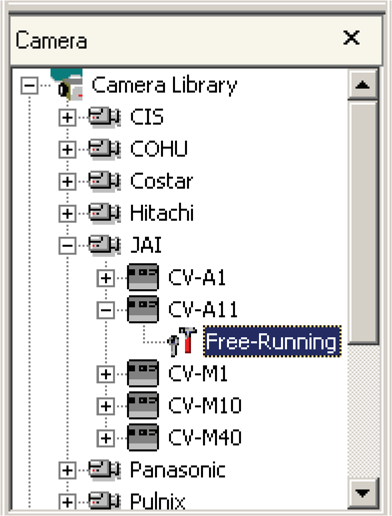
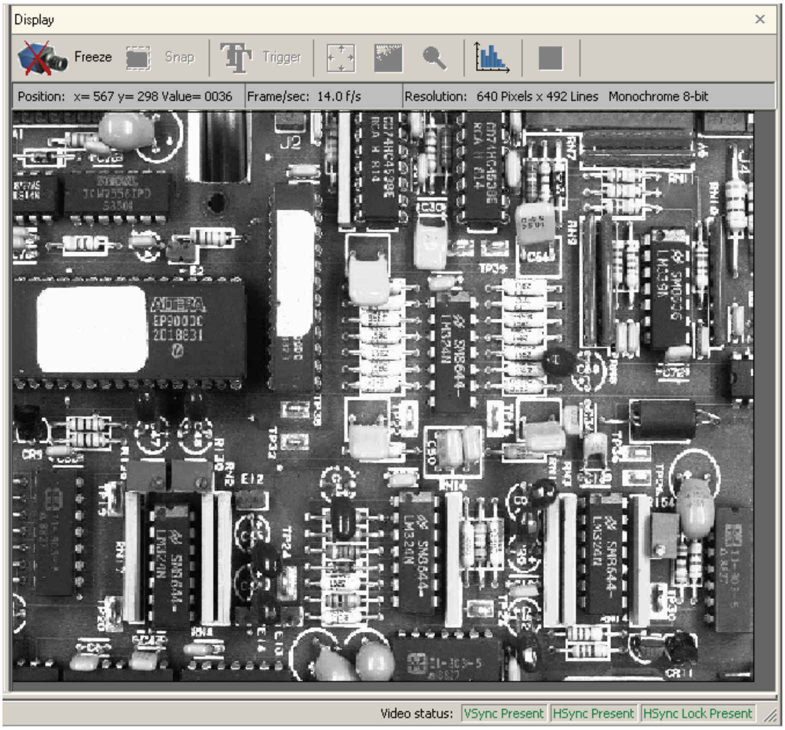


Figure 43: CamExpert: Selecting a camera file

- Assuming that the camera is connected to the X64-AN Quad on channel one, click the CamExpert grab button. Adjust the camera iris and focus. The following screen image shows CamExpert in grab mode (with the mouse pointer ready to stop the capture). During live grab CamExpert displays the video frame rate and resolution.



Position: x= 567 y= 298 Value= 0036 Frame/sec: 14.0 f/s Resolution: 640 Pixels x 492 Lines Monochrome 8-bit

Video status: VSync Present HSync Present HSync Lock Present

Figure 44: CamExpert: Live acquisition window

- If you have multiple cameras connected to the X64-AN Quad, select each board server in turn to verify live grab from each camera.
- Use the **File-Save** dialog to save this camera configuration file (*.ccf) with user entered information. CamExpert provides information for each field based on the file originally loaded. Modify the fields such as camera mode and board configuration, to describe the parameter setup. Modify the file name and click Save. The ccf file is located in the default Sapera user folder.

Overview of Basic Timing Parameters

CamExpert only shows parameters applicable to the acquisition board and camera type. When configuring parameters for a new camera start by selecting or entering the basic horizontal timing parameters and pixel clock frequency as defined by the camera manufacturer.

Following is a brief overview of the basic timing parameters required for interfacing any camera.

- **Video Standard**: An analog camera is either RS170/NTSC or CCIR/PAL or Non Standard. The group of Non Standard cameras covers all cameras that are not one of the basic TV standards. Video capture of such cameras is only limited by the capabilities of the frame grabber hardware used.
- **Image Sensor readout**: The JAI CV-A11 output is full frame progressive scan video. Basic TV standard video is interlaced.

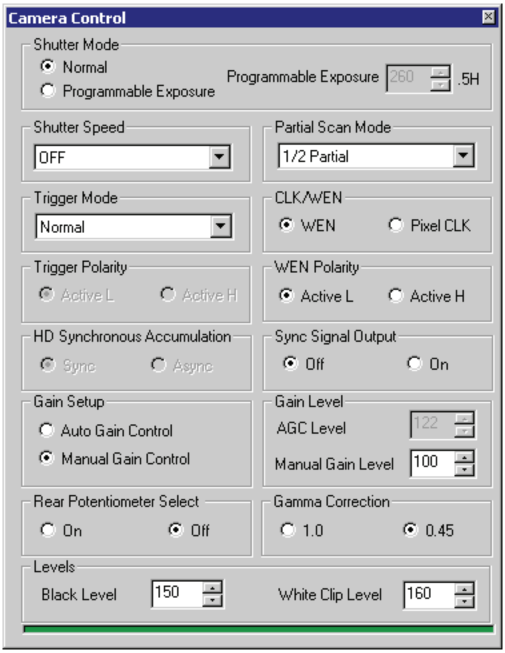
- **Horizontal and Vertical Active**: Defines the frame resolution of the camera. These parameters along with the front porch/back porch/sync values define the camera timing parameters. When a new camera is interface to a Sapera frame grabber, these values are entered based on the camera specifications.
- **Pixel Clock Frequency**: Set the sampling clock frequency the frame grabber board will use to digitize the analog video. The camera specifications will define the pixel clock required. Typically this will produce square pixels.
- **Video Sync Source**: Defines if H/V sync is embedded with the video or is on separate signals.

step 2: JAI CV-A11 in Partial Scan Mode

The JAI CV-A11 partial scan mode increases the camera frame rate by decreasing the number of video lines output to the frame grabber (refer to the camera manual for information). CamExpert is easily configured to support each JAI CV-A11 partial scan mode, independently of whether the camera is in free run or trigger mode.

JAI CV-A11 Control Setting

Using partial scan mode requires only two setup changes. One to the camera control tool and the second to a CamExpert parameter. The following screen image shows the JAI CV-A11 tool setting the scan parameter to ½ **Partial** (240 video lines).



Camera Control

Shutter Mode: ● Normal ○ Programmable Exposure — Programmable Exposure 260 .5H

Shutter Speed: OFF

Partial Scan Mode: 1/2 Partial

Trigger Mode: Normal

CLK/WEN: ● WEN ○ Pixel CLK

Trigger Polarity: ● Active L ○ Active H

WEN Polarity: ● Active L ○ Active H

HD Synchronous Accumulation: ● Sync ○ Async

Sync Signal Output: ● Off ○ On

Gain Setup: ○ Auto Gain Control ● Manual Gain Control

Gain Level: AGC Level 122; Manual Gain Level 100

Rear Potentiometer Select: ○ On ● Off

Gamma Correction: ○ 1.0 ● 0.45

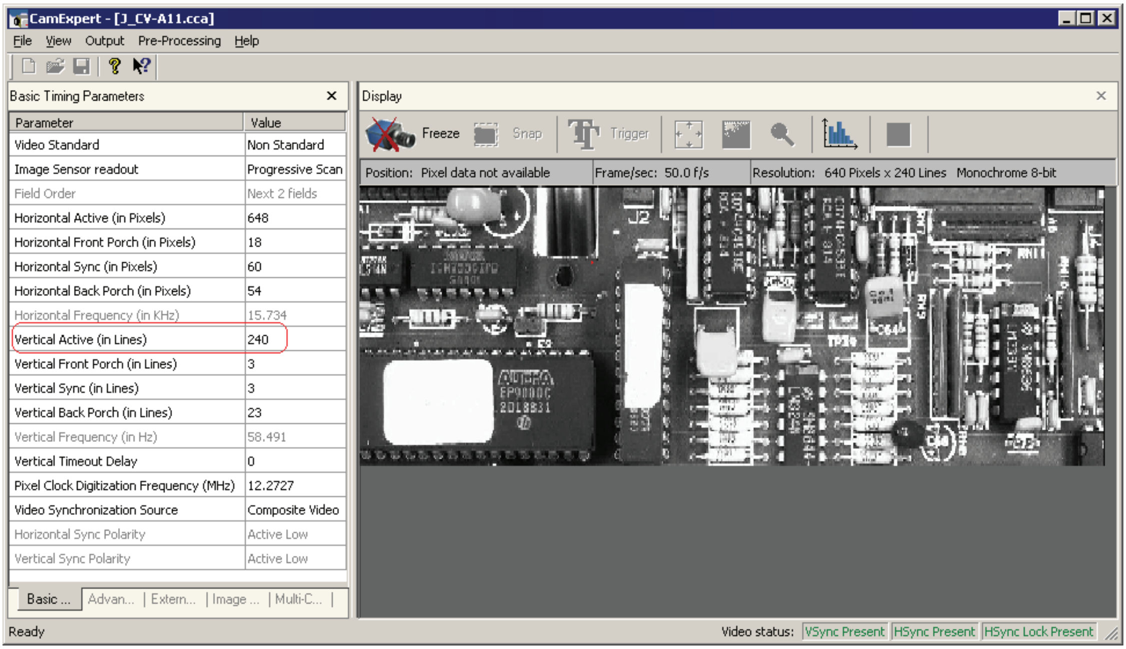
Levels: Black Level 150; White Clip Level 160

Figure 45: Setting the JAI CV-A11 to Partial Scan Mode

CamExpert Setting

The following screen image shows the CamExpert basic timing parameter for **Vertical Active** set to 240 lines, matching the JAI CV-A11 video output. The test image grab in the display window shows the capture of the JAI CV-A11 partial scan output from the center portion of its CCD (refer to the camera manual for more information on all partial scan modes).

CamExpert can be configure to capture whatever number of video lines output by the camera. Modified CamExpert settings should be saved as new camera configuration files with an appropriate description and unique file name.



Parameter	Value
Video Standard	Non Standard
Image Sensor readout	Progressive Scan
Field Order	Next 2 fields
Horizontal Active (in Pixels)	648
Horizontal Front Porch (in Pixels)	18
Horizontal Sync (in Pixels)	60
Horizontal Back Porch (in Pixels)	54
Horizontal Frequency (in KHz)	15.734
Vertical Active (in Lines)	240
Vertical Front Porch (in Lines)	3
Vertical Sync (in Lines)	3
Vertical Back Porch (in Lines)	23
Vertical Frequency (in Hz)	58.491
Vertical Timeout Delay	0
Pixel Clock Digitization Frequency (MHz)	12.2727
Video Synchronization Source	Composite Video
Horizontal Sync Polarity	Active Low
Vertical Sync Polarity	Active Low

Position: Pixel data not available | Frame/sec: 50.0 f/s | Resolution: 640 Pixels x 240 Lines Monochrome 8-bit

Ready

Video status: VSync Present | HSync Present | HSync Lock Present

Figure 46: CamExpert: Partial Scan Live Grab

step 3: JAI CV-A11 in Edge Pre-select (trigger) Mode

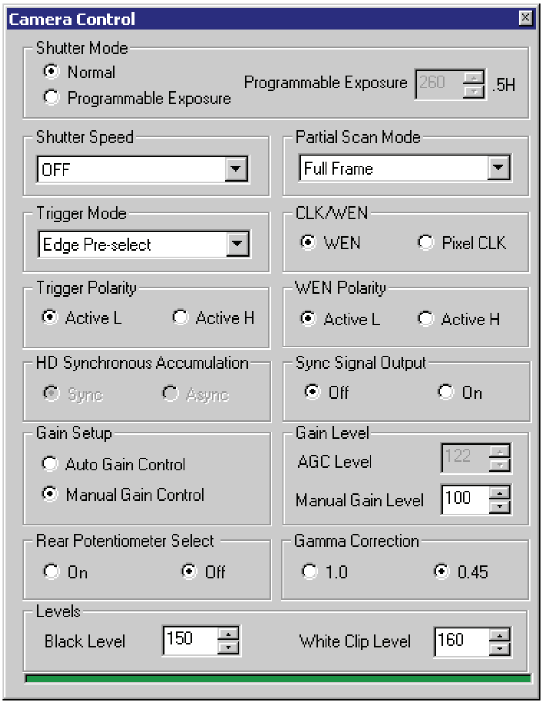
The JAI CV-A11 supports a number of trigger modes. This example describes the setup for Edge Pre-select. As described in the camera user manual, the leading edge of the trigger pulse initiates the exposure. The trigger pulse to the camera is sent and controlled by the frame grabber to ensure correct timing with the camera sync. In this mode, the CCD exposure time is simply controlled by the camera setup.

The frame grabber receives an external signal connected to its external trigger input. This is typically some asynchronous event used to signal the frame grabber to initiate and acquire 1 to n video frames from the camera. The type of external trigger signal is dependent on the capabilities of the frame grabber hardware.

For setup and testing purposes CamExpert provides a control button to simulate the asynchronous event trigger. This software trigger control simplifies the camera configuration and testing procedure before the vision system is placed into its final location.

JAI CV-A11 Control Setting

The following screen image shows the camera setup for Edge Pre-select trigger which is selected from the Trigger Mode drop menu. The shutter mode can either be normal or programmed exposure. The scan mode can be full or partial as required.



Camera Control

Shutter Mode
● Normal
○ Programmable Exposure
Programmable Exposure 260 .5H

Shutter Speed: OFF

Partial Scan Mode: Full Frame

Trigger Mode: Edge Pre-select

CLK/WEN: ● WEN ○ Pixel CLK

Trigger Polarity: ● Active L ○ Active H

WEN Polarity: ● Active L ○ Active H

HD Synchronous Accumulation: ● Sync ○ Async

Sync Signal Output: ● Off ○ On

Gain Setup
○ Auto Gain Control
● Manual Gain Control

Gain Level
AGC Level 122
Manual Gain Level 100

Rear Potentiometer Select: ○ On ● Off

Gamma Correction: ○ 1.0 ● 0.45

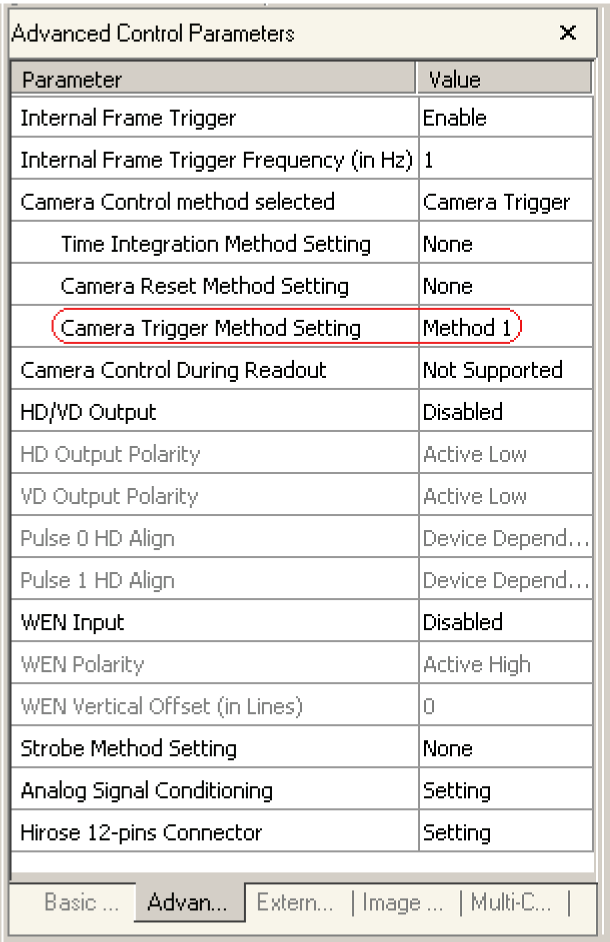
Levels
Black Level 150
White Clip Level 160

Figure 47: Setting the JAI CV-A11 to a Trigger Mode

CamExpert Settings

CamExpert parameters are in two groups—those which select the control method and those which enable one of the possible control methods.

- First select a Sapera camera trigger method that matches the trigger timing required by the camera. In this example with the JAI CV-A11, Sapera camera trigger method 1 is selected.

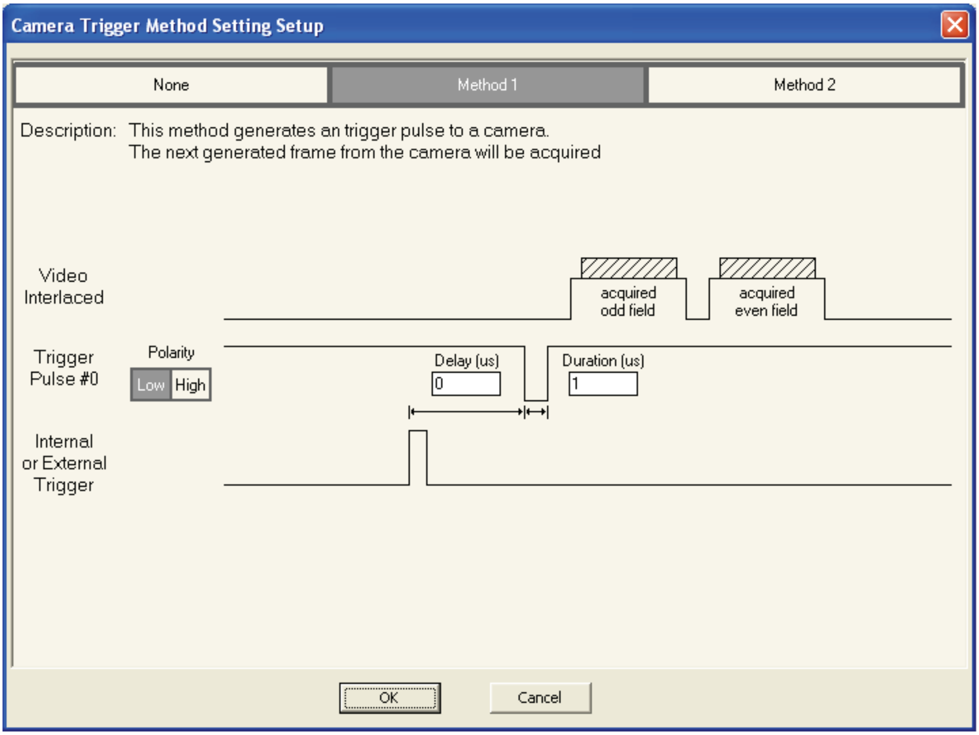


Advanced Control Parameters

Parameter	Value
Internal Frame Trigger	Enable
Internal Frame Trigger Frequency (in Hz)	1
Camera Control method selected	Camera Trigger
Time Integration Method Setting	None
Camera Reset Method Setting	None
Camera Trigger Method Setting	Method 1
Camera Control During Readout	Not Supported
HD/VD Output	Disabled
HD Output Polarity	Active Low
VD Output Polarity	Active Low
Pulse 0 HD Align	Device Depend...
Pulse 1 HD Align	Device Depend...
WEN Input	Disabled
WEN Polarity	Active High
WEN Vertical Offset (in Lines)	0
Strobe Method Setting	None
Analog Signal Conditioning	Setting
Hirose 12-pins Connector	Setting

Figure 48: CamExpert: Setting Trigger Mode

- Next step is to program the trigger method parameters such as polarity, delay and duration. CamExpert provides a graphical menu to enter those variables which must be defined as required by the camera specifications. In this example with the JAI CV-A11, the trigger duration is set to 600µs to meet the camera specifications defined in its user manual.


Camera Trigger Method Setting Setup

Description: This method generates an trigger pulse to a camera.
The next generated frame from the camera will be acquired

Delay (us): 0
Duration (us): 1

Figure 49: CamExpert: Camera Trigger Method 1 Timing

- The desired Sapera control method now must be enabled. This parameter is required because a frame grabber board and camera may support different acquisition or exposure methods. When multiple methods are supported and setup, only one can be the controlling method at any one time. In this example, with two control methods having been defined, the **Enable Control Method** drop menu selects which control method to activate.

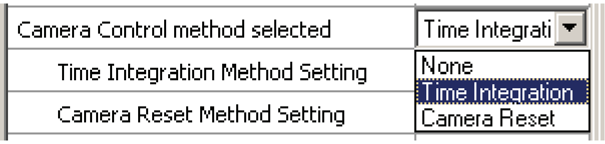
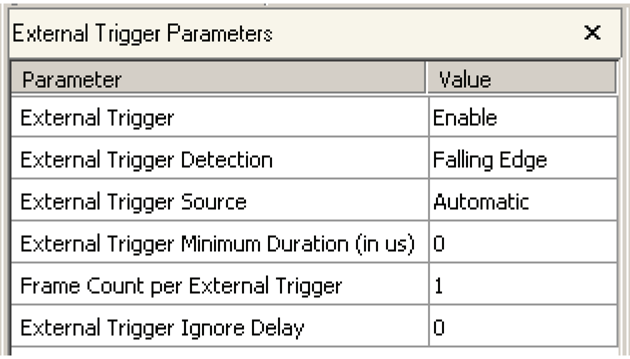

Camera Control method selected: Time Integrati
Time Integration Method Setting
Camera Reset Method Setting

Figure 50: CamExpert: Advanced Control Parameters

- With the camera control method defined and enabled, the last configuration item is to define the external trigger signal to the frame grabber. The CamExpert External Trigger Parameters tab, shown in the following screen image, has external trigger enabled, with detection set to falling edge, and a trigger minimum duration as 0µs (edge detection—not level detection). The last parameter defines the number of sequential video frames captured on a single external trigger event.



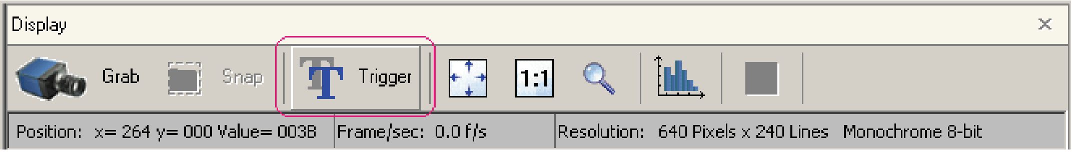
External Trigger Parameters

Parameter	Value
External Trigger	Enable
External Trigger Detection	Falling Edge
External Trigger Source	Automatic
External Trigger Minimum Duration (in us)	0
Frame Count per External Trigger	1
External Trigger Ignore Delay	0

Figure 51: CamExpert: External Trigger Parameters

Testing the Setup

CamExpert provides a control button to emulate an asynchronous event trigger which simplifies testing the camera configuration before the vision system is placed into its final location. When all camera and CamExpert parameters are set, click on the Trigger button to initiate video acquisition. With the setup example described a sequence of 15 frames are captured and displayed in the CamExpert display window.



Position: x= 264 y= 000 Value= 003B

Frame/sec: 0.0 f/s

Resolution: 640 Pixels x 240 Lines Monochrome 8-bit

Figure 52: CamExpert: Software Trigger Button

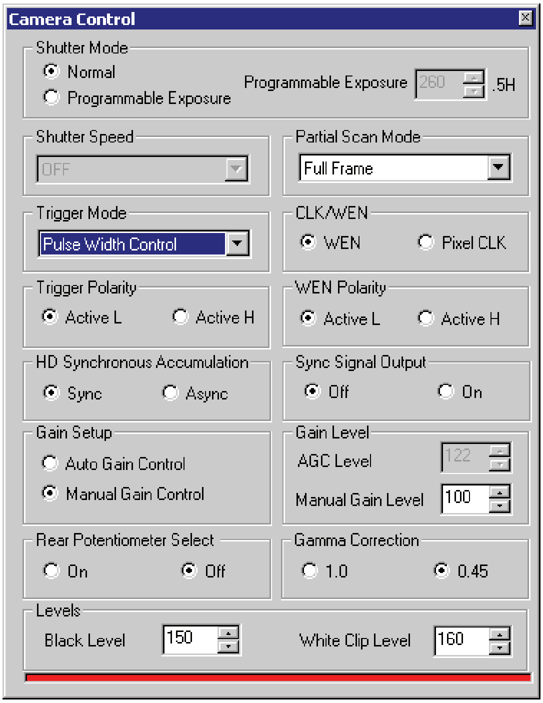
Modified and tested CamExpert settings, from this example or any custom configuration, should be saved as a new camera configuration file with an appropriate description and unique file name. Loading the saved camera file, either in CamExpert again or from your custom vision application, is the simplest method to configure Sapera for the camera used.

step 4: JAI CV-A11 in Time Integration Mode

The JAI CV-A11 trigger mode Pulse Width Control is a single pulse time integration mode. The leading edge of the trigger initiates the video capture while the trigger pulse width determines the exposure time.

JAI CV-A11 Control Setting

The following screen image shows the camera setup for Pulse Width Control trigger which is selected from the Trigger Mode drop menu. Camera shutter speed is not applicable in this mode.



Camera Control

Shutter Mode
● Normal
○ Programmable Exposure
Programmable Exposure 260 .5H

Shutter Speed: OFF

Partial Scan Mode: Full Frame

Trigger Mode: Pulse Width Control

CLK/WEN: ● WEN ○ Pixel CLK

Trigger Polarity: ● Active L ○ Active H

WEN Polarity: ● Active L ○ Active H

HD Synchronous Accumulation: ● Sync ○ Async

Sync Signal Output: ● Off ○ On

Gain Setup
○ Auto Gain Control
● Manual Gain Control

Gain Level
AGC Level 122
Manual Gain Level 100

Rear Potentiometer Select: ○ On ● Off

Gamma Correction: ○ 1.0 ● 0.45

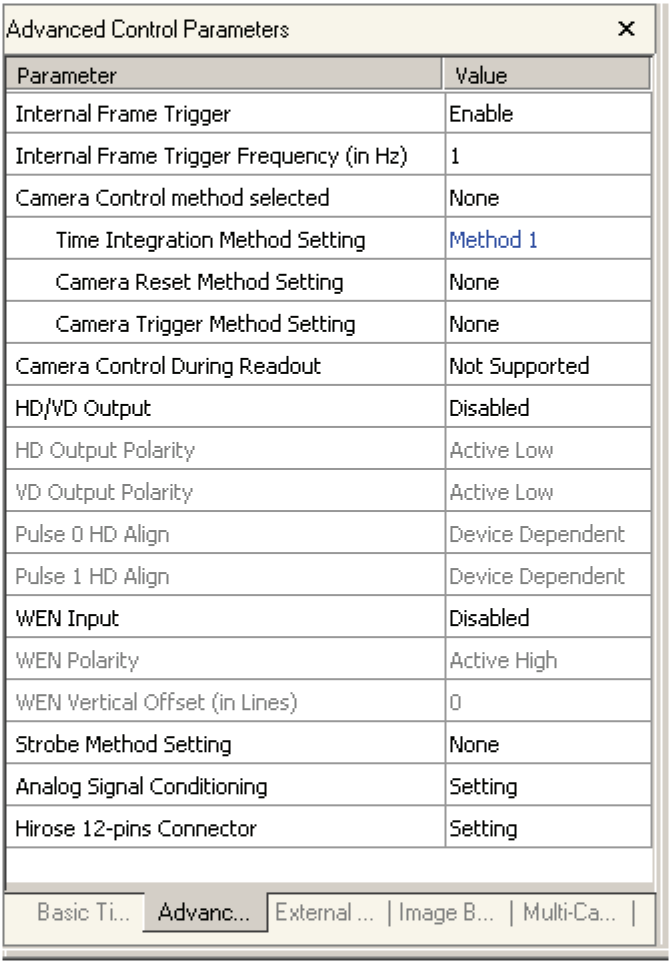
Levels
Black Level 150
White Clip Level 160

Figure 53: CamExpert: Pulse Width Trigger Setup

CamExpert Settings

CamExpert parameters are in two groups similar to the previous example. Those which select the control method and those which enable one of the possible control methods.

- First select a Sapera time integration method that matches the timing and control required by the camera. In this example with the JAI CV-A11, time integration method 1 is selected.

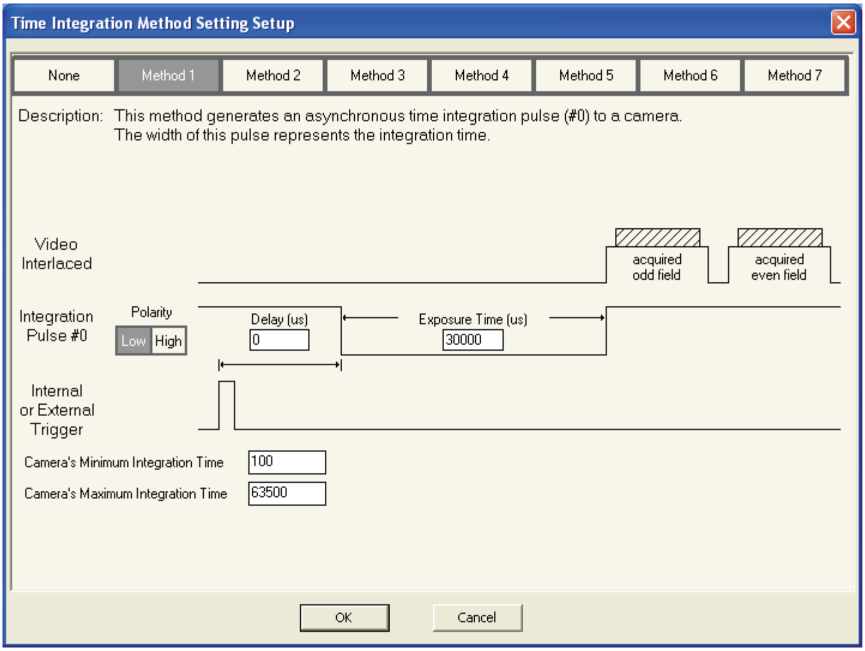


Advanced Control Parameters

Parameter	Value
Internal Frame Trigger	Enable
Internal Frame Trigger Frequency (in Hz)	1
Camera Control method selected	None
Time Integration Method Setting	Method 1
Camera Reset Method Setting	None
Camera Trigger Method Setting	None
Camera Control During Readout	Not Supported
HD/VD Output	Disabled
HD Output Polarity	Active Low
VD Output Polarity	Active Low
Pulse 0 HD Align	Device Dependent
Pulse 1 HD Align	Device Dependent
WEN Input	Disabled
WEN Polarity	Active High
WEN Vertical Offset (in Lines)	0
Strobe Method Setting	None
Analog Signal Conditioning	Setting
Hirose 12-pins Connector	Setting

Figure 54: CamExpert: Time Integration Method Selection

- Next step is to program the integration method parameters such as polarity, delay and exposure time. CamExpert provides a graphical menu to enter those variables which must be defined as required by the camera specifications and exposure desired. In this example with the JAI CV-A11, the exposure duration was set to 30000µs simply to have a proper exposure at the test bench.



Description: This method generates an asynchronous time integration pulse (#0) to a camera. The width of this pulse represents the integration time.

Delay (us): 0

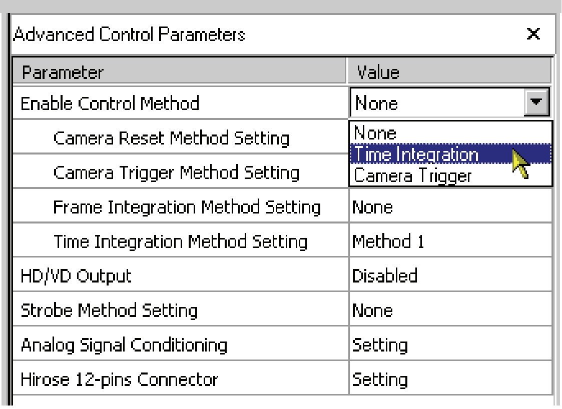
Exposure Time (us): 30000

Camera's Minimum Integration Time: 100

Camera's Maximum Integration Time: 63500

Figure 55: CamExpert: Time Integration Method Timing

- The desired Sapera control method now must be enabled. This parameter is required because a frame grabber board and camera may support different acquisition or exposure methods. When multiple methods are supported and setup, only one can be the controlling method at any one time. In this example, with two control methods having been defined, the **Enable Control Method** drop menu selects which control method to activate.



Advanced Control Parameters

Parameter	Value
Enable Control Method	None
Camera Reset Method Setting	None Time Integration Camera Trigger
Camera Trigger Method Setting	
Frame Integration Method Setting	None
Time Integration Method Setting	Method 1
HD/VD Output	Disabled
Strobe Method Setting	None
Analog Signal Conditioning	Setting
Hirose 12-pins Connector	Setting

Figure 56: CamExpert: Enable Time Integration

Testing the Setup

With the time integration setup click on the CamExpert Grab button to have a live acquisition in the display window. Time integration can work either in free run or external trigger mode.

Modified and tested CamExpert settings, from this example or any custom configuration, should be saved as a new camera configuration file with an appropriate description and unique file name. Loading the saved camera file, either in CamExpert again or from your custom vision application, is the simplest method to configure Sapera for the camera used.

Note on Analog Camera Timing Relationships

For analog cameras, the following formulas show the relationship between the PCLK parameter and the Horizontal and Vertical total. These values must be accurate if the acquisition board drives the synchronization signals to the camera (board is in Master Mode).

- The HS and VS signal frequencies are:

$$\boxed{\frac{1}{HS_{freq}} = \frac{1}{PCLK_{freq}} * H_{total}} \qquad \boxed{\frac{1}{VS_{freq}} = \frac{1}{HS_{freq}} * V_{total}}$$

Overview of Video Bandwidth and System Limitations

Some high resolution/high frame rate cameras can output more data than can be transferred through the host computer's PCI bus. A successful imaging application must account for the camera data bandwidth and possibly control frame rate or image resolution to bring the bandwidth requirements to within the system's limitations.

Bandwidth is defined in two different ways. Peak bandwidth is the highest data rate occurring at any time during the data transfer. The average bandwidth is the amount of data per unit of time being transferred.

Each is calculated as follows:

- Peak Bandwidth (MBps) = Pixel Clock Frequency *Bpp *nb channels
- Average Bandwidth (MBps) = Frame Width *Frame Height *Frame Rate *Bpp
  where:
  MBps = Mega-Bytes per second
  Bpp = Bytes per pixel
  nb = number of …

When the bandwidth required by the frame grabber exceeds the capacity of the PCI 32/64-bit bus, the following techniques can reduce and optimize the average bandwidth.

Bandwidth Reduction Techniques

- A linear relationship exists between the average bandwidth required and the acquisition image height. For example, if four cameras of 1K x 1K, at some frame rate, need to transfer 160MB per/second of data, those four cameras at a 512 x 1K resolution will only need a bandwidth of 80MB per/second, which is now within the capability of the PCI-32 bus.

Bandwidth Optimization Techniques

The following techniques are suggestions for applications that require all possible optimizations from the host system. Implementing these suggestions require a thorough understanding of your computer system setup and its BIOS controls.

- Allocate a separate IRQ for the frame grabber. BIOS settings can be used to manually assign the IRQ number to a particular PCI slot. As an alternative, the Window Device Manager can be used to force a specific IRQ to a specific PCI slot.
- Minimize the PCI latency timer in the BIOS setting; the value is given in CLK cycles.
- Use a high-performance AGP VGA card to decrease the image display system overhead when live acquisition is required.
- Avoid any hard drive write/read operations and network access through PCI LAN interfaces during intensive image transfers by the frame grabber.

Important: Some computer systems do not provide the BIOS controls described. Review your system manual.

Sapera Grab Demo Example

Program	**Start • All Programs • Teledyne DALSA • Sapera LT • Demos • Grab Demo**
Program file	**…\Sapera\Demos\Classes\vc\GrabDemo\Release\GrabDemo.exe**
Workspace	**…\Sapera\Demos\Classes\vc\SapDemos.dsw**
Description	This program demonstrates the basic acquisition functions included in the Sapera library. The program allows you to acquire images, either in continuous or in one-shot mode, while adjusting acquisition parameters. The program code can be extracted for use within your own application.
Remarks	Grab Demo was built using Visual C++ 6.0 by means of the MFC library and is based on the Sapera standard API and Sapera C++ classes. See the Sapera User's and Reference manuals for further information.

Using the Grab Demo

Server Selection

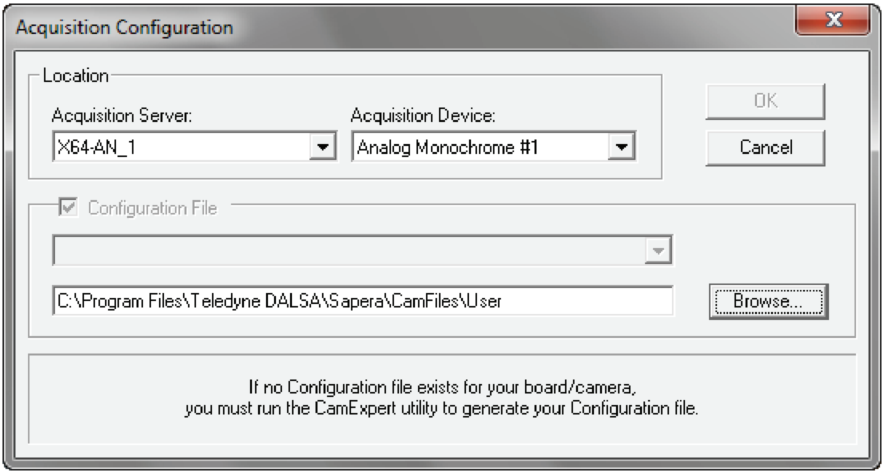
Run Grab Demo from: **Start • All Programs • Teledyne DALSA • Sapera LT • Demos • Frame Grabbers • Grab Demo**.

When activated, Grab Demo first displays the "Acquisition Configuration" window. The first drop down menu allows you to select any installed Sapera acquisition server (that is, installed Teledyne DALSA acquisition hardware using Sapera drivers). The second drop down menu allows you to select the available input devices present on the selected server.

CCF File Selection

The "Acquisition Configuration" window is also used to select the camera configuration file required for the connected camera. Sapera camera files contain timing parameters and video conditioning parameters. The default folder used for camera configuration files is also used by the CamExpert utility to save user generated or modified camera files.

Use Sapera CamExpert to generate the camera configuration file based on the timing and control parameters entered (see "Using Sapera CamExpert with X64-AN Quad" on page 78 for examples). The CamExpert live acquisition window allows immediate verification of the parameters. CamExpert reads both Sapera *.cca and *.cvi files for backwards compatibility with the original Sapera legacy camera files.



Acquisition Server: X64-AN_1

Acquisition Device: Analog Monochrome #1

☑ Configuration File

C:\Program Files\Teledyne DALSA\Sapera\CamFiles\User

If no Configuration file exists for your board/camera,
you must run the CamExpert utility to generate your Configuration file.

Figure 57: Camera File Selection Menu

Grab Demo Main Window

The main window provides control buttons and a central region where the grabbed image is displayed. Developers can use the source code supplied with the demo as a foundation to quickly create and test the desired imaging application.

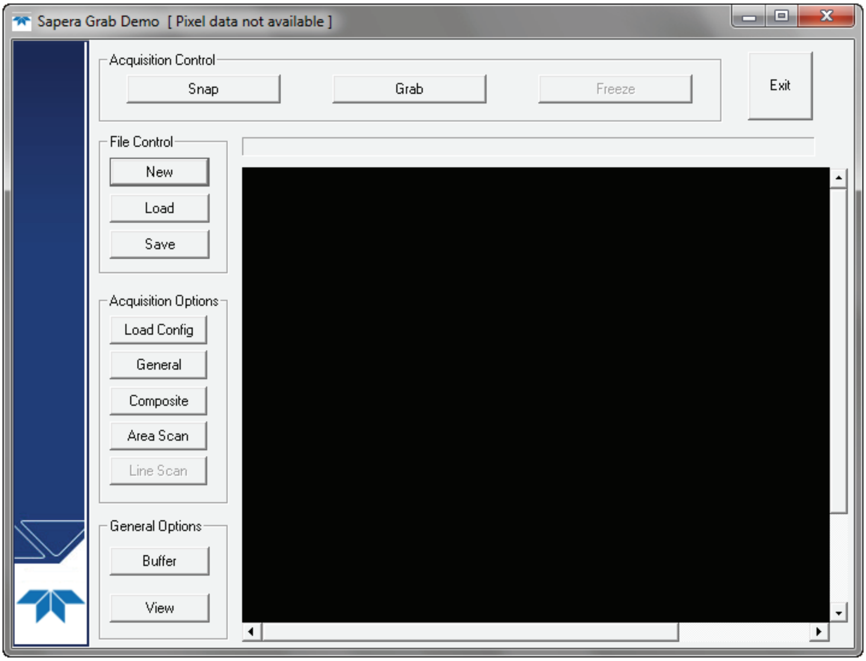


Figure 58: Sapera Grab Demo Program

The various functions are described below:

File Control

Three controls are provided for image file transfers

- **New:** Clears the current image frame buffer.
- **Load:** Retrieves images in BMP, TIF, CRC, JPG, and RAW formats.
- **Save:** Prompts for a file name, file save location, and image format.

Acquisition Options

Note that unsupported functions are grayed out and not selectable. Function support is dependent on the frame grabber hardware in use.

- **General – Acquisition Settings:** Allows for X64-AN Quad external trigger mode enabling.
- **Area Scan – Camera Control:** Provides trigger, reset, and integrate controls when supported by the current hardware and driver. Also offers master HS and VS output.
- **Line Scan – Camera Control:** This dialog is not applicable to the area scan frame grabber.
- **Composite - Conditioning:** Offers Brightness and Contrast controls.
- **Load Config:** Opens the dialog window Acquisition Configuration allowing the user to load a new set of camera files. This is the same window displayed when the Sapera Grab Demo is first started.

Acquisition Control

- **Grab:** Displays live digitized video from your video source. If your source is a camera, focus and adjust the lens aperture for the best exposure. Use a video generator as a video source to acquire reference images.
- **Freeze:** Stops live grab mode.
  The grabbed image can be saved to disk via the **File Control • Save** control.
- **Snap:** A single video frame is grabbed.
- **Abort** Exits the current grab process immediately. If any video signal problem prevents the freeze function from ending the grab, click **Abort**.

General Options

Note: functions grayed out are not supported by acquisition hardware.

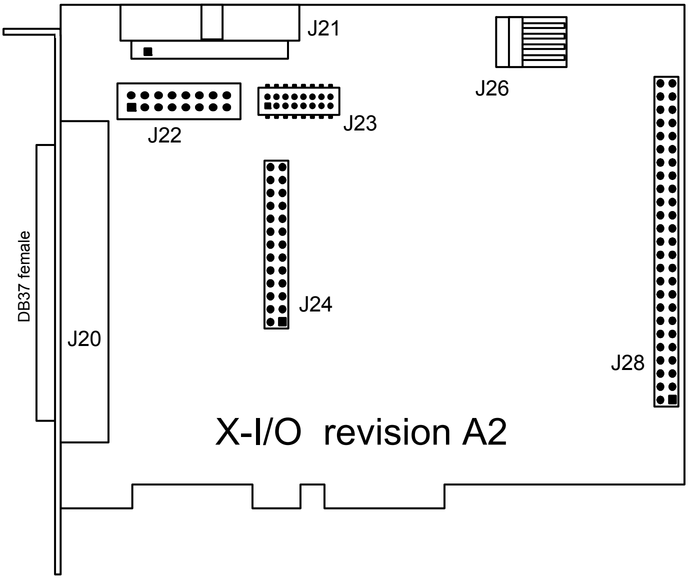
- **Buffer:** Select from supported frame buffer counts, size, and types.
- **Count and Size**: Select the number of frame buffers and the image size here.
- Type – **Contiguous**: Frame buffers are allocated in contiguous system memory (single memory block - no segmentation).
- Type – **Scatter-Gather**: (Preferred) Frame buffers are allocated throughout system memory in noncontiguous memory (paged pool). Pages are locked in physical memory so a scatter-gather list can be constructed. This type allows for the allocation of very large size buffers or large buffer counts.
- Type – **Off-screen Video**: The buffer is allocated in off-screen video memory and uses the display adapter hardware to perform a fast copy from video memory to video memory.
- Type – **Overlay**: The frame buffer is allocated in video memory where the display adapter overlay hardware uses color-keying to view the overlay buffer.
- **Format**: Shows frame buffer pixel formats as supported by the hardware and camera files used.

Appendix: X-I/O Module Option

X-I/O Module Overview

- The X-I/O module requires X64-AN Quad board driver version 1.10 (or later) and Sapera LT version 5.30 (or later).
- Occupies an adjacent slot to the X64-AN Quad. Slot can be either PCI-32 or PCI-64—no PCI signals or power are used.
- Connects to the X64-AN Quad via a 16 pin flat ribbon cable OC-IO0C-ANLVDS). J23 on X-I/O to J16 on X64-AN Quad.
- X-I/O provides 8 outputs software selectable as NPN (current sink) or PNP (source driver) type drivers. See "Outputs in NPN Mode: Electrical Details" on page 103 and "Outputs in PNP Mode: Electrical Details" on page 104.
- X-I/O provides 2 opto-coupled inputs. See "Opto-coupled Input: Electrical Details" on page 105.
- X-I/O provides 6 TTL level inputs with software selectable transition point. See "TTL Input Electrical Details" on page 105.
- X-I/O provides both +5 volt and +12 volt power output pins on the DB37, where power comes directly from the host system power supply.
- Onboard flash memory to store user defined power up I/O states.

X-I/O Module Connector Location



X-I/O Connector List

J20	DB37 female I/O signals connector.
J23	16 pin miniature header connector (connect to X64-AN Quad J16 via supplied ribbon cable OC-IO0C-ANLVDS).
J21, J22, J24, J28	Reserved.
J26	Connect PC power via floppy drive power cable.

X-I/O Module Installation

Grounding Instructions: Static electricity can damage electronic components. Please discharge any static electrical charge by touching a grounded surface, such as the metal computer chassis, before performing any hardware installation. If you do not feel comfortable performing the installation, please consult a qualified computer technician. **Never** remove or install any hardware component with the computer power on.

Board Installation

Installing an X-I/O Module to an existing X64-AN Quad installation takes only a few minutes. Install the X-I/O board into the host system as follows:

- Power off the computer system that has the installed X64-AN Quad board.
- Insert the X-I/O module into any free PCI slot (no PCI electrical connections are used), securing the bracket.
- Connect the X-I/O module 16 pin ribbon cable from J23 to the X64-AN Quad board J16.
- Power on the computer again.
- For new X64-AN Quad and X-I/O module installations, simply follow the procedure to install Sapera and the X64-AN Quad driver (see "Installation" on page 9).

X64-AN Quad and X-I/O Driver Update

- If both Sapera 5.30 and X64-AN Quad driver 1.10 need to be installed, follow the procedure "Upgrading Sapera or any Teledyne DALSA Board Driver" on page 17. This procedure steps through the upgrade of both Sapera and the board driver—typically required when installing the X-I/O module in the field.
- If the X64-AN Quad installation already has the required Sapera and board driver version, install the X-I/O module and perform a firmware update as described in "Executing the Firmware Loader from the Start Menu" on page 12.

X-I/O Module External Connections to the DB37

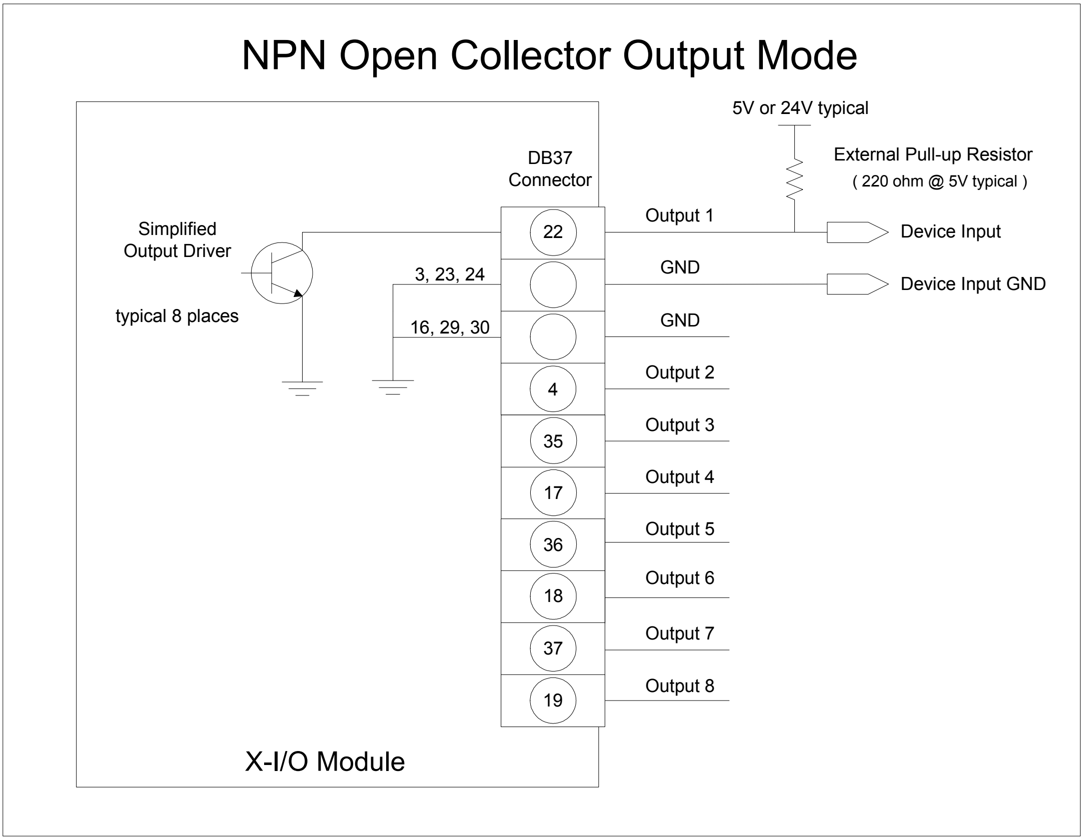
Users can assemble their interface cable, using some or all of the signals available on the X-I/O module DB37. Use a male DB37 with thumb screws for a secure fit. Wiring type should meet the needs of the imaging environment.

DB37 Pinout Description

Pin #	Signal	Description
1	IN_OPTO_1+	Input #1 (Opto-coupled)
20	IN_OPTO_1-	
2	IN_OPTO_2+	Input #2 (Opto-coupled)
21	IN_OPTO_2-	
3, 23, 24	Gnd	
22	OUT_TTL_1	output #1
4	OUT_TTL_2	output #2
5	USER_PWR	Power for the TTL Outputs in PNP mode
6, 7, 8, 9, 10, 11		Reserved
25, 26, 27, 28		Reserved
16, 29, 30	Gnd	
12	Power	PC +5V (1A max)
31	Power	PC +12V (1A max)
13	IN_TTL_3	Input 3 (TTL)
32	IN_TTL_4	Input 4 (TTL)
14	IN_TTL_5	Input 5 (TTL)
33	IN_TTL_6	Input 6 (TTL)
15	IN_TTL_7	Input 7 (TTL)
34	IN_TTL_8	Input 8 (TTL)
35	OUT_TTL_3	output 3
17	OUT_TTL_4	output 4
36	OUT_TTL_5	output 5
18	OUT_TTL_6	output 6
37	OUT_TTL_7	output 7
19	OUT_TTL_8	output 8

Outputs in NPN Mode: Electrical Details

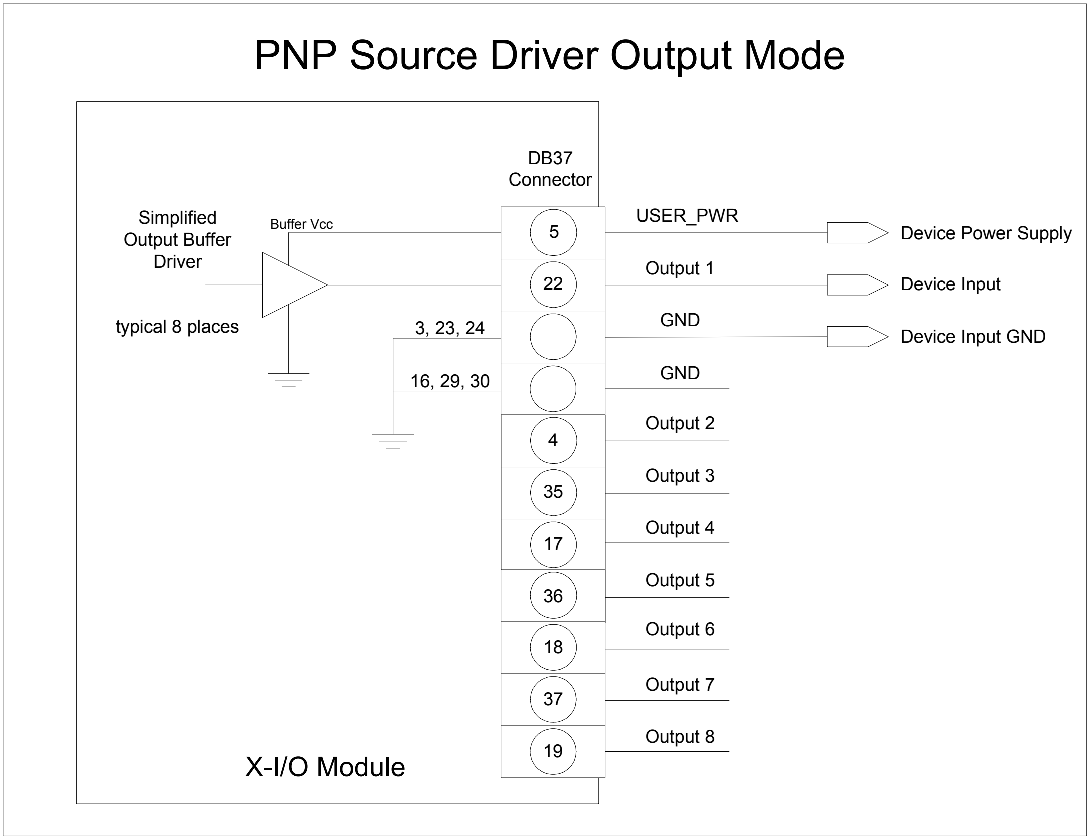
When the outputs are configured for NPN mode (open collector - sink mode) the user is required to provide an external input pull-up resistor on the signal being controlled by the X-I/O output. A simplified schematic and important output specifications follow:



- Each output can sink 700 mA.
- Over-current thermal protection will automatically shut down the output device.

Outputs in PNP Mode: Electrical Details

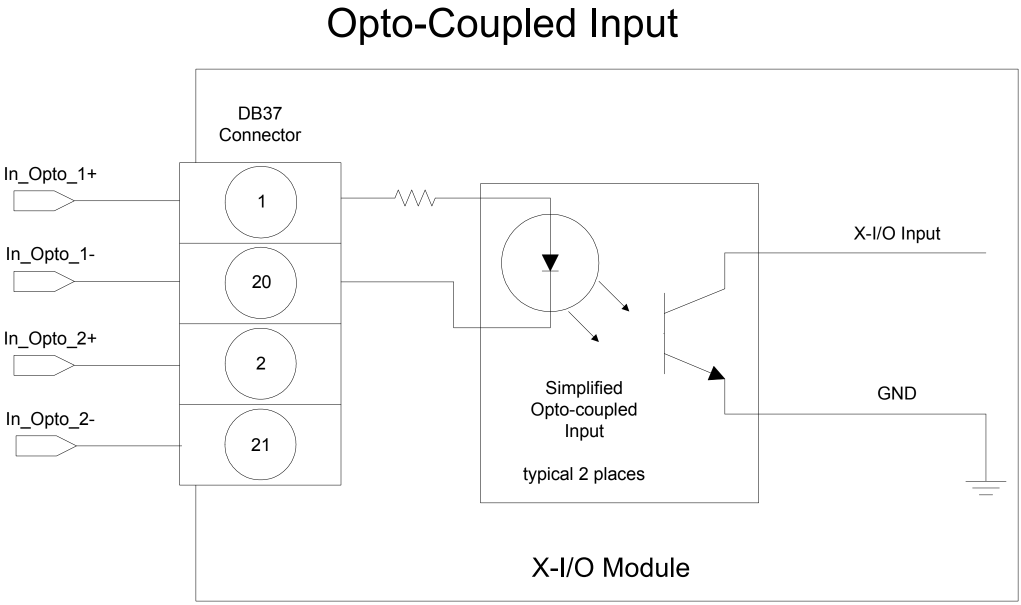
When the outputs are configured for PNP mode (source driver) the user is required to provide the output supply voltage (USR_PWR). A simplified schematic and important output specifications follow:



- User provides the output power supply voltage (7 volts to 35 volts).
- Maximum source driver output current is 350 mA.
- Source driver with over-current protection (all outputs will shut down simultaneously). The over-current fault circuit will protect the device from short-circuits to ground with supply voltages of up to 35V.

Opto-coupled Input: Electrical Details

The two opto-coupled inputs can be used either with TTL or RS422 sources. A simplified input schematic and important electrical specifications are listed below.



Input reverse breakdown voltage	5 volts minimum
Maximum average forward input current	25 mA
Maximum input frequency	200 kHz
Maximum Sapera call-back rate	System processing dependent

TTL Input Electrical Details

The six TTL inputs are software configurable (see "Configuring User Defined Power-up I/O States" on page 106) for standard TTL logic levels or industrial logic systems (typically 24 volts). The design switch points are as follows:

- TTL level mode: trip point at 2V +/- 5%
- Industrial level mode: trip point at 16V +/- 5%

X-I/O Module Sapera Interface

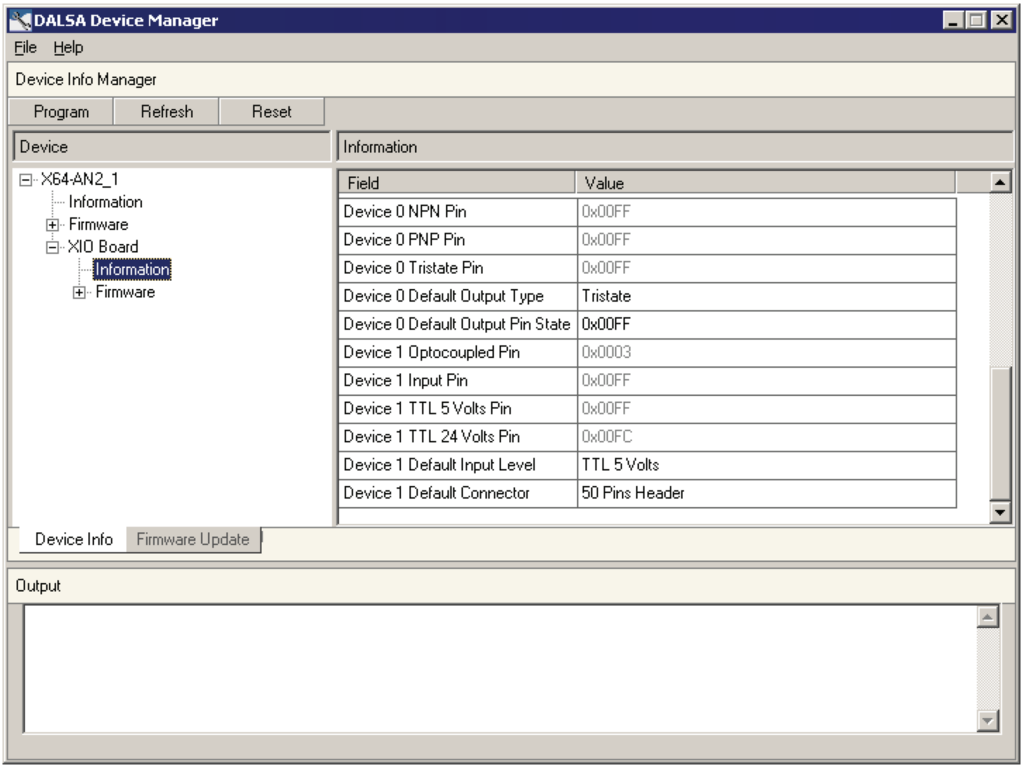
Sapera version 5.30 (or later) provides support for the X-I/O module via an I/O class and demonstration program. Users can use the demonstration program as is, or use the demo program source code to implement X-I/O controls within the custom imaging application.

This section describes configuring the X-I/O module power up state, using the X-I/O demo program, and describes the Sapera Class to program and read the X-I/O module along with sample code.

Configuring User Defined Power-up I/O States

The X-I/O module power up state is stored onboard in flash memory. User configuration of this initial state is performed by the Device Manager program. Run the program via the windows start menu: (**Start • Programs • Teledyne DALSA • X64-AN Quad Device Driver • Device Manager**).

The Device Manager provides information on the installed X64-AN Quad board and its firmware. With an X-I/O module installed, click on **XIO Board – Information**, as shown in the following figure.



Field	Value
Device 0 NPN Pin	0x00FF
Device 0 PNP Pin	0x00FF
Device 0 Tristate Pin	0x00FF
Device 0 Default Output Type	Tristate
Device 0 Default Output Pin State	0x00FF
Device 1 Optocoupled Pin	0x0003
Device 1 Input Pin	0x00FF
Device 1 TTL 5 Volts Pin	0x00FF
Device 1 TTL 24 Volts Pin	0x00FC
Device 1 Default Input Level	TTL 5 Volts
Device 1 Default Connector	50 Pins Header

The XIO information screen shows the current status of **Device 0**—the output device, and **Device 1**—the input device. A few items are user configurable for X-I/O board power up state. Click on the item to display a drop list of available capabilities, as described below.

- **Device 0 – Default Output Type**
  choose Tristate mode (i.e. output disconnected), or PNP mode, or NPN mode.
- **Device 0 – Default Output Pin State**
  A window is displayed to select a logic low or high state for each output pin. Click on each pin that should be logic high by default.
- **Device 1 – Default Input Level**
  Select the input logic level as TTL 5 Volts or 24 Volts, dependent on the signal type being input to the X-I/O module.
- **Device 1 – Default Connector**
  DB37 is the supported output connector, as described in this section.

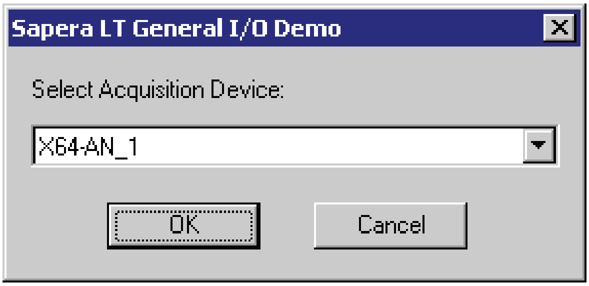
Programming the User Configuration

After changing any user configurable X-I/O mode from the factory default state, click on the **Program** button (located on the upper left), to write the new default state to flash memory. The Device Manager message output window will display "Successfully updated EEProm". The program can now be closed.

Using Sapera LT General I/O Demo

The Sapera General I/O demo program controls the I/O capabilities of the X-I/O module on the Sapera board product. The demo will present to the user only the controls pertaining to the selected hardware (in the case of multiple installed boards).

Run the demo via the windows start menu: (**Start • Programs • Teledyne DALSA • Sapera LT • Demos • General I/O Demo**). The first menu presents a drop list of all installed Sapera Acquisition Devices with I/O capabilities. In the following figure the X64-AN Quad board is selected. Click OK to continue.



Select Acquisition Device: X64-AN_1

General I/O Module Control Panel

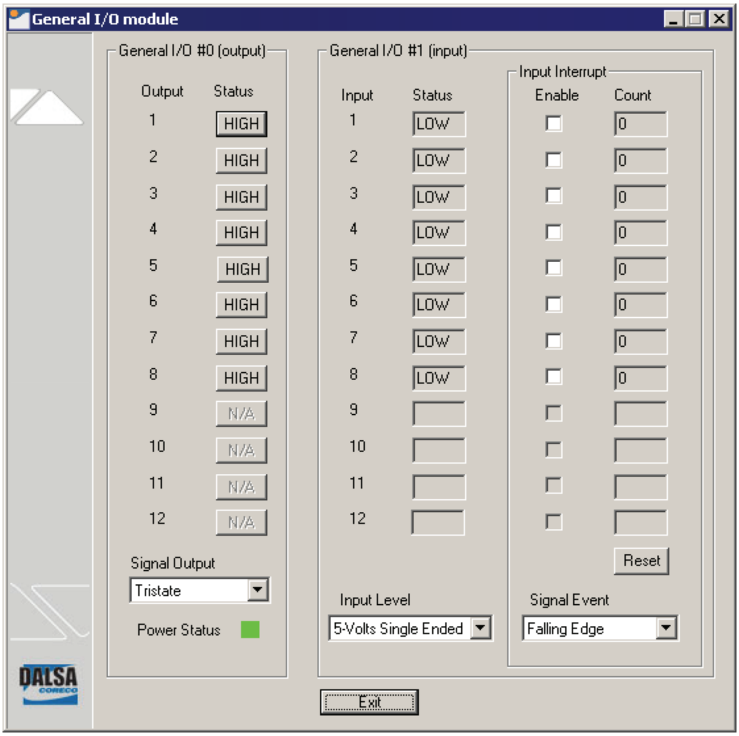
The I/O module control demo presents the I/O capabilities of the installed hardware. The following figure shows the X-I/O module connected to the X64-AN Quad board.

Output Pins: The first column displays the current state of the eight output pins (I/O Device #0).

- The startup default state is user configured using the Device Manager program.
- The state of each output can be changed by clicking on its status button.
- Use the Signal Output drop menu to select the output mode (Tristate, PNP, NPN).

Input Pins: The second section provides input pin status (I/O device #1). Note that this program is a demo, therefore no action takes place on an input event.

- The first column reads the logic level present on each input. The Input Level drop menu changes the logic level from 5V TTL to 24V logic. Use the Device Manager program to select the default logic level type.
- The second column demonstrates activating interrupts on individual inputs. In this demo program, use the Enable box to activate the interrupt on an input. The Count box will tally detected input events. Use the Signal Event drop menu to select which input signal edge to detect. The Reset button clears all event counts.



General I/O module

General I/O #0 (output)

Output	Status
1	HIGH
2	HIGH
3	HIGH
4	HIGH
5	HIGH
6	HIGH
7	HIGH
8	HIGH
9	N/A
10	N/A
11	N/A
12	N/A

Signal Output: Tristate

Power Status: 🟩

General I/O #1 (input)

		Input Interrupt	
Input	Status	Enable	Count
1	LOW	□	0
2	LOW	□	0
3	LOW	□	0
4	LOW	□	0
5	LOW	□	0
6	LOW	□	0
7	LOW	□	0
8	LOW	□	0
9		□	
10		□	
11		□	
12		□	

Input Level: 5-Volts Single Ended

Signal Event: Falling Edge

Sapera LT General I/O Demo Code Samples

The following source code was extracted from the General I/O demo program. The comments highlight the areas that an application developer needs for embedding X-I/O module controls within the imaging application.

Main I/O Demo code

```
BOOL CGioMainDlg::OnInitDialog()
{

[ . . . ]

// some declarations
UINT32  m_gioCount;
int m_ServerIndex;
int m_ResourceIndex;


// Show the Server Dialog to select the acquisition device
CGioServer dlg(this);
if (dlg.DoModal() == IDOK)
{

    m_ServerIndex = dlg.GetServerIndex();
    m_ServerName = dlg.GetServerName();

    if ( m_ServerIndex != -1)
        {
        // Get the number of resources from SapManager for ResourceGio type by using
        // - the server index chosen in the dialog box
        // - the resource type to enquire for Gio

    m_gioCount=SapManager::GetResourceCount(m_ServerIndex,SapManager::ResourceGio);

        // Create all objects [see the function following]
        if (!CreateObjects()) { EndDialog(TRUE); return FALSE; }

        [ . . . ]

        //Loop for all resources
        for (UINT32 iDevice = 0; (iDevice < MAX_GIO_DEVICE) && (iDevice < m_gioCount);
                                  iDevice++)
        {
            [ . . . ]

            // direct read access to low-level Sapera C library capability to check
            // I/O Output module
            if (m_pGio[iDevice]->IsCapabilityValid(CORGIO_CAP_DIR_OUTPUT))
            status = m_pGio[iDevice]->GetCapability(CORGIO_CAP_DIR_OUTPUT,&capOutput);
```

```
           // direct read access to low-level Sapera C library capability to
           // check I/O Input module
           if (m_pGio[iDevice]->IsCapabilityValid(CORGIO_CAP_DIR_INPUT))
           status = m_pGio[iDevice]->GetCapability(CORGIO_CAP_DIR_INPUT,&capInput);

           [ . . . ]
           // Constructor used for I/O Output module dialog.
           if (capOutput)
           {
           m_pDlgOutput[iDevice] = new CGioOutputDlg(this, iDevice, m_pGio[iDevice]);
           }

           [ . . . ]

           // Constructor used for I/O Input module dialog.
           if (capInput)
           {
           m_pDlgInput[iDevice] = new CGioInputDlg(this, iDevice, m_pGio[iDevice]);
           }
        }  //end for
}   // end if

[ . . . ]
}
```

Function CreateObjects()

```
BOOL CreateObjects()
{
CWaitCursor wait;

// Loop for all I/O resources
for (UINT32 iDevice = 0; (iDevice < MAX_GIO_DEVICE) && (iDevice < m_gioCount);
        iDevice++)
{
    // The SapLocation object specifying the server where the I/O resource is located
    SapLocation location(m_ServerIndex, iDevice);

        // The SapGio constructor is called for each resource found.
        m_pGio[iDevice] = new SapGio(location);

        // Creates all the low-level Sapera resources needed by the I/O object
        if (m_pGio[iDevice] && !*m_pGio[iDevice] && !m_pGio[iDevice]->Create())
        {
            DestroyObjects();
            return FALSE;
        }
    }
return TRUE;
}
```

Output Dialog: CGioOutputDlg class (see Sapera Gui class)

```
void CGioOutputDlg::UpdateIO()
{
    UINT32 output=0;
    UINT32 state=0;
    BOOL status;
    [ . . . ]

    // We loop to get all I/O pins.
    for (UINT32 iIO=0; iIO < (UINT32)m_pGio->GetNumPins(); iIO++)
    {
        [ . . . ]

        // We set the current state of the current I/O pin by using
        // - the pin number on the current I/O resource
        // - the pointer to pin state
        //  ( SapGio ::PinLow if low and SapGio ::PinHigh if high)
        status = m_pGio->SetPinState(iIO, (SapGio::PinState)state);
    }

}
```

Input Dialog: CGioInputDlg class. (see Sapera Gui class)

```
BOOL CGioInputDlg::Update()
{
    SapGio::PinState state = SapGio::PinState::PinLow;
    BOOL status = true;
    UINT32 iIO;
    UINT32 jIO;

    if (m_pGio == NULL)
        return FALSE;

    // We loop to get all I/O pins.
    for (iIO=0; iIO < (UINT32)m_pGio->GetNumPins(); iIO++)
    {
        m_pGio->SetDisplayStatusMode(SapManager::StatusLog, NULL);
        // We get the current state of the current I/O pin by using
        // the pin number on the current I/O resource
        // the pointer to pin state
        //  ( SapGio ::PinLow if low and SapGio ::PinHigh if high)

        status = m_pGio->GetPinState(iIO, &state);
        m_pGio->SetDisplayStatusMode(SapManager::StatusNotify, NULL);

        [ . . . ]
    }

    [ . . . ]
}
```

I/O Event Handling

```
void CGioInputDlg::GioCallbackInfo(SapGioCallbackInfo *pInfo)
{
CGioInputDlg* pInputDlg;
CString strEventCount;

    // We get the application context associated with I/O events
    pInputDlg = (CGioInputDlg*)pInfo->GetContext();

    // We get the current count of I/O events
    strEventCount.Format("%d", pInfo->GetEventCount());

    // We get the I/O pin number that generated an I/O event and apply the changes.
    pInputDlg->m_GioEventCount[pInfo->GetPinNumber()]++;
}
```

TELEDYNE DALSA
A Teledyne Technologies Company

Contact Information

Sales Information

Visit our web site: www.teledynedalsa.com/mv

Email: mailto:info@teledynedalsa.com

Canada

Teledyne DALSA — Montreal office
7075 Place Robert-Joncas, Suite #142
St. Laurent, Quebec, Canada
H4M 2Z2

Tel: (514) 333-1301
Fax: (514) 333-1388

USA Sales

Teledyne DALSA — Billerica office
700 Technology Park Drive
Billerica, Ma.
01821

Tel: (978) 670-2000
Fax: (978) 670-2010

Asia Sales

Teledyne DALSA Asia Pacific
Ikebukuro East 13F
3-4-3 Higashi Ikebukuro,
Toshima-ku, Tokyo
Japan

Tel: +81 3 5960 6353
Fax: +81 3 5960 6354

European Sales

Teledyne DALSA Europe
Breslauer Str. 34
D-82194 Gröbenzell (Munich)
Germany

Tel: +49 - 8142 – 46770
Fax: +49 - 8142 - 467746

Technical Support

Submit any support question or request via our web site:

Technical support form via our web page: Support requests for imaging product installations, Support requests for imaging applications Camera support information Product literature and driver updates	http://www.teledynedalsa.com/mv/support

Glossary of Terms

ADC

Analog-to-Digital conversion is an electronic process in which a continuously variable (analog) signal is changed, without altering its essential content, into digital data.

Bandwidth

Describes the measure of data transfer capacity. A computer system's PCI expansion bus is rated for a maximum peak data bandwidth of 132 MB/s. PCI devices must share the maximum PCI bus bandwidth when transferring data to and from system memory or other devices.

Bus

A common pathway, or channel, between multiple devices. Besides the computer's internal bus to memory and system components, peripheral buses such as PCI and AGP, allow adding or changing devices that make up the computer system.

Clamper Circuit

A circuit that establishes a fixed level for the video signal at the beginning of each scanning line.

CMRR

Common-Mode Rejection Ratio: The ratio of the common-mode noise or interference voltage at the input of a circuit, to the corresponding noise or interference voltage at the output.

Composite Video

A video signal that is composed of the luminance and color information plus the synchronization signals together. Common composite video formats are NTSC and PAL.

Contiguous Memory

A block of physical memory occupying consecutive addresses.

Driver

Also called a device driver, a program routine that links a peripheral device to the operating system. Specific to the Bandit-II, its VGA driver is required for its display adapter functionality and a device driver is required for its frame grabber capabilities.

Frame

One complete image data set or its equivalent storage space.

Frame buffer

An area of memory used to hold a frame of image data. A frame buffer may exist on the acquisition hardware or be allocated by the acquisition hardware device driver in host system memory.

Genlock

When two cameras are genlocked, their internal sync circuits are driven by a common external source. These cameras output video frames synchronous to each other.

Grab

Acquiring an image frame by means of a frame grabber.

Grayscale

In image processing, the range of available brightness levels, displayed in shades of gray. In an 8-bit system, the gray scale contains values from 0 to 255.

Host

Refers to the computer system that supports the installed frame grabber.

Interlaced

Describing the standard television method of raster scanning in which the image is the product of two fields, each of which is made up of the image's alternate lines (i.e., one field is comprised of lines 1, 3, 5, etc., and the other is comprised of lines 2, 4, 6, etc.)

Low Pass Filter

A filter that blocks high frequencies and allows lower frequencies to pass through. Used to limit undesirable analog information (such as high frequency video noise) before converting to digital data.

NTSC

National Television Systems Committee. Color TV standard used in North America and other countries. The interlaced video signal is composed of a total of 525 video lines at a frame rate of 30 Hz.

PAL

Phase Alteration by Line. Color TV standard used in most of Europe and other countries. The interlaced video signal is composed of a total of 625 video lines at a frame rate of 25 Hz.

PCI

Peripheral Component Interconnect. The PCI local bus is a 32-bit high performance expansion bus intended for interconnecting add-in boards, controllers, and processor/memory systems.

Pixel

A contraction of "picture element". The number of pixels describes the number of digital samples taken of the analog video signal. The number of pixels per video line by the number of active video lines describes the acquisition image resolution. The binary size of each pixel (e.g., 8-bits, 15-bits, 24-bits) defines the number of gray levels or colors possible for each pixel.

PLL

Short for phase-locked loop. A PLL ensures that a signal is locked on a specific frequency.

Progressive Scan Camera

The progressive scan format outputs data from the camera (the signal) in sequential order as it is scanned. The scan format produces a full frame of video in a continuous stream, rather than half the image per output sequence in standard interlaced cameras.

Scatter Gather

Host system memory allocated for frame buffers is virtually contiguous but physically scattered throughout all available memory.

Trigger

A mechanism that initiates an action when an event occurs such as synchronizing an image acquisition to an external event. A trigger generally causes a program routine to be executed such as the resetting of camera exposure and/or the firing of a strobe light.

UART

A UART (Universal Asynchronous Receiver/Transmitter) is the microchip with programming that controls an interface to its attached serial devices.

WEN

Write ENable: Output by some video cameras to indicate valid video. Also indicates frame timing when the camera does not use VS.

Index